RAPID LIFT AND DRAG PREDICTION TOOL WITH MULTI-LAYER PERCEPTRON MODEL FOR SUPERCRITICAL AIRFOILS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

CANER ATLI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

JANUARY 2023

Approval of the thesis:

RAPID LIFT AND DRAG PREDICTION TOOL WITH MULTI-LAYER PERCEPTRON MODEL FOR SUPERCRITICAL AIRFOILS

submitted by **CANER ATLI** in partial fulfillment of the requirements for the degree of **Master of Science in Mechanical Engineering, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. M. A. Sahir Arıkan
Head of the Department, **Mechanical Engineering** ________________

Assist. Prof. Dr. Özgür Uğraş Baran
Supervisor, **Mechanical Engineering, METU** ________________

Assoc. Prof. Dr. Hande Alemdar
Co-Supervisor, **Computer Engineering, METU** ________________

Examining Committee Members:

Prof. Dr. Oğuz Uzol
Aerospace Eng, METU ________________

Assist. Prof. Dr. Özgür Uğraş Baran
Mechanical Engineering, METU ________________

Assoc. Prof. Dr. Hande Alemdar
Computer Engineering, METU ________________

Assist. Prof. Dr. Onur Baş
Mechanical Engineering, TED University. ________________

Assist. Prof. Dr. Ali Karakuş
Mechanical Engineering, METU ________________

Date: ...

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Last name : Atlı, Caner

Signature :

ABSTRACT

RAPID LIFT AND DRAG PREDICTION TOOL WITH MULTI-LAYER PERCEPTRON MODEL FOR SUPERCRITICAL AIRFOILS

Atlı, Caner
Master of Science, Mechanical Engineering
Supervisor: Asst. Prof. Dr. Özgür Uğraş Baran
Co-Supervisor: Assoc. Prof. Dr. Hande Alemdar

January 2023, 74 pages

In the early stages of aircraft design, selecting the shape of parts such as wings and tails can be time-consuming. Typically, tools that can provide rapid predictions of aerodynamic coefficients are used in these stages to reduce the duration of the preliminary design phase. However, available tools and research cannot provide accurate and reliable solutions in transonic and fully turbulent flows. A new tool is developed to provide aerodynamic properties such as drag, lift, maximum lift, and drag divergence Mach number by utilizing machine learning methodologies. The tool consist of a reliable shape descriptor for airfoil shape based on Class Shape Function Transformation. Then a Multilayer Perceptron machine learning method is developed using high-accuracy CFD solution database and airfoil shape descriptions. Then, new airfoil shapes and aerodynamic data is produced consistently and accurately in the transonic and turbulent flow regime. In the end, we have achieved a solution accuracy with an R^2 score of 0.9935 for validation set lift coefficient using the new tool. Predicted aerodynamic coefficients are drag coefficient, lift coefficient and drag divergence Mach number. The tool provides solutions for 27000 cases within seconds or angle of attack range of -2° to 8°.

Keywords: Deep Learning, Rapid Predictions, Drag Divergence, Preliminary Design

ÖZ

ÇOK KATMANLI ALGILAYICI İLE EĞİTİLMİŞ SÜPERKRİTİK KANATLAR İÇİN HIZLI TASARIM ARACI

Atlı, Caner
Yüksek Lisans, Makina Mühendisliği
Tez Yöneticisi: Asst. Prof. Dr. Özgür Uğraş Baran
Ortak Tez Yöneticisi: Asst. Prof. Dr. Hande Alemdar

Ocak 2023, 74 sayfa

Uçak tasarımının ilk aşamalarında, kanat ve kuyruk gibi parçaların şeklini seçmek zaman alıcı olabilir. Tipik olarak, ön tasarım aşamasının süresini azaltmak için bu aşamalarda aerodinamik katsayıların hızlı tahminlerini sağlayabilen araçlar kullanılır. Ancak mevcut araçlar ve araştırmalar, transonik ve tamamen türbülanslı akışlarda doğru ve güvenilir çözümler sağlayamaz. Makine öğrenimi metodolojileri kullanılarak sürükleme, kaldırma, maksimum kaldırma ve ıraksama Mach sayısı gibi aerodinamik katsayıları sağlamak için yeni bir araç geliştirildi. Bu araç, Sınıf Şekil Fonksiyon Dönüşümüne dayalı olarak kanat kesit şekli için güvenilir bir şekil tanımlayıcısından oluşur. Daha sonra, yüksek doğruluklu CFD çözüm veritabanı ve kanat şekli dönüşümleri kullanılarak bir Çok Katmanlı Perceptron makine öğrenimi yöntemi geliştirildi. Ardından, transonik ve türbülanslı akış rejiminde tutarlı ve doğru bir şekilde yeni kanat kesit şekilleri ve aerodinamik veriler üretildi. Sonunda, bu aracı kullanarak R^2 puanı 0,9935 olan bir çözüm doğruluğu elde ettik. Öngörülen aerodinamik katsayılar, sürükleme katsayısı, kaldırma katsayısı ve ıraksama Mach sayısıdır. Araç, saniyeler içinde 27000 vaka sunabilir ve -2° ila 8° hücum açısı aralığında çözümler sunabilir.

Anahtar Kelimeler: Derin Öğrenme, Anlık Tahmin, Açılma Mach Sayısı, Ön Tasarım

Dedication

To the shoulders of the giants that we stand upon.

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisor Asst. Prof. Dr. Özgür Uğraş Baran and co-supervisor Assoc. Prof. Dr. Hande Alemdar for their guidance, advice, criticism, encouragements and insight throughout the research.

The author would also like to thank Dr. Cihat Duru and Fazıl Selçuk Gömeç for their suggestions and comments.

The author wishes to express his gratitude to his family, his friends and gubu for always supporting him and standing by him.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vi

ACKNOWLEDGMENTS ix

TABLE OF CONTENTS x

LIST OF TABLES xiii

LIST OF FIGURES xiv

LIST OF ABBREVIATIONS xvi

LIST OF SYMBOLS xvii

CHAPTERS

1 INTRODUCTION 1

1.1 Aim and scope of this study 3

1.2 Contents of this study 4

2 REQUIREMENTS FOR ACCURATE AIRFOIL PERFORMANCE ESTIMATIONS 5

2.1 Flow and shape parameters that affect airfoil performance 5

2.2 Common Airfoil Shape Representations 6

2.3 Transonic Flow around the Airfoils 8

2.3.1 Drag Divergence Definition 9

2.4 Physical parameters that affect airfoil performance 10

3 COMMON SHAPE PARAMETRIZATION METHODS 13

3.1 Interpolation 13

3.2 Beziér Curves 15

3.3 Class-Shape Function Transformation 16

3.3.1 Legacy Airfoil Representation with CST.. 17

4 RAPID PREDICTION TOOLS IN AIRFOIL PREDICTIONS 23

4.1 Low-Fidelity Commercial Tools 23

4.1.1 DATCOM 23

4.1.2 XFOIL 24

4.1.3 Vortex Lattice Method 24

4.2 Neural Network Methods 24

4.2.1 Artificial Neural Networks 25

4.2.2 Convolutional Neural Networks 26

4.3 Conclusion.. 28

5 NEURAL NETWORK STUDIES FOR THE PREDICTION MODEL 31

5.1 Problem definition.. 31

5.2 Multilayer Perceptron.. 33

5.3 Prediction model structure 37

6 DATA GENERATION 47

6.1 Mesh independence 47

6.2 Validation study 50

7 NEURAL NETWORK MODEL FOR COEFFICIENT PREDICTIONS IN TRANSONIC REGIME 53

7.1 Dataset definition 53

7.2 Model construction and results 56

8 CONCLUSION AND FUTURE WORKS 67

REFERENCES 69

APPENDICES

A. Model Usage 73

B. CST Script with Python 74

LIST OF TABLES

TABLES

Table 2.1 An example Selig format airfoil 7

Table 4.1 Review summary of some studies in the literature 29

Table 5.1 An except from the dataset 32

Table 5.2 MLP construction excerpt... 34

Table 5.3 Learning rate with Keras excerpt.. 36

Table 5.4 Performance metrics for different model complexities and transformation complexities 38

Table 5.5 Parametrization method comparison for lift coefficient 42

Table 6.1 Flow conditions for mesh independence.. 47

Table 6.2 Validation experiment conditions 50

Table 6.3 Solution parameters used in this study 51

Table 7.1 Flow conditions for supercritical CFD runs 54

Table 7.2 Validation set metrics 57

Table 9.1 Example output of the model... 73

Table 9.2 Model output example 73

Table 9.3 Python script for CST parametrization 74

LIST OF FIGURES

FIGURES

Figure 1.1 Wing cross-section of a generic airliner 2
Figure 2.1 Angle of attack definition 5
Figure 2.2 A typical pressure distribution over the airfoil 6
Figure 2.3 Drag vs. Mach graph of SC2-0714 airfoil..9
Figure 2.4 Effect of maximum thickness on drag, taken from [6] 11
Figure 2.5 Effect of maximum thickness on the maximum lift, taken from [6]......11
Figure 3.1 Curvature graph of an interpolated airfoil...14
Figure 3.2 Series of expansion and compression waves across an interpolated airfoil surface...14
Figure 3.3 Beziér curve representation of NACA 64-012a airfoil 16
Figure 3.4 Fig.2 of reference 17
Figure 3.5 Contribution of each shape function to the transformation of SC2-0714 airfoil 19
Figure 3.6 7th order transformation of the SC2-0714 legacy airfoil........................20
Figure 3.7 Error plot for CST of SC2-0714 airfoil..20
Figure 5.1 MLP visualization 33
Figure 5.2 Typical overfit graph...35
Figure 5.3 Learning rate scheduler effect on training 36
Figure 5.4 Structure of the MLP network 37
Figure 5.5 Training history of lift coefficient model...37
Figure 5.6 Loss vs. transformation complexity...39
Figure 5.7 R2 score vs. transformation complexity 39
Figure 5.8 Loss vs. model complexity...40
Figure 5.9 R2 score vs. model complexity..41
Figure 5.10 Correlation plots of the MLP model 42
Figure 5.11 Correlation plot for test set lift coefficient...43

Figure 5.12 Lift coefficient comparison between prediction and CFD for randomly selected test airfoils 45
Figure 6.1 Mach number contour of mesh independence solutions 48
Figure 6.2 Grid dependence results. 48
Figure 6.3 Farfield radius study results.. 49
Figure 6.4 Comparison of the NASA website (left) and Flowpsi (right) 49
Figure 6.5. Pressure coefficient graphs of two solvers and experiment results...... 51
Figure 6.6 Close up view of the final mesh 52
Figure 7.1 RAE2822 CFD solutions.. 55
Figure 7.2 Mach number contour of RAE2822, 0.84 Mach and 0° angle of attack freestream.. 56
Figure 7.3 Drag coefficient training graphs... 56
Figure 7.4 Lift coefficient training graphs... 57
Figure 7.5 Correlation plots of the validation set... 58
Figure 7.6 MaDD comparison for a validation airfoil.. 59
Figure 7.7 Predictions for a validation set airfoil... 60
Figure 7.8 Test set correlation plots.. 61
Figure 7.9 Test set results for MaDD 63
Figure 7.10 Comparison plot with experimental results [31] 65
Figure 7.11 Comparison plot with experimental results [32] 66

LIST OF ABBREVIATIONS

POD	Proper Orthogonal Decomposition
CFD	Computational Fluid Dynamics
NURBS	Non-uniform Rational B-Splines
RMSE	Root mean squared error
MSE	Mean squared error
CAD	Computer-Aided Design
CST	Class-shape function transformation
AoA	Angle of attack
RANS	Reynolds-Averaged Navier Stokes
MAE	Mean Absolute Error

LIST OF SYMBOLS

V	Magnitude of relative wind speed
a	Speed of sound
ρ	Fluid density
L	Reference length
S_{ref}	Reference area
μ	Dynamic viscosity
Re	Reynolds number
Ma	Mach number

CHAPTER 1

INTRODUCTION

The airfoil is a critical element in aerodynamics, as it is the defining shape for many vital aircraft components. These include wings, which are responsible for providing lift and enabling flight, and control surfaces, such as flaps and ailerons, allowing the pilot to control the aircraft's roll, pitch, and yaw [1]. An example of a wing cross section can be seen in Figure 1.1. The airfoil also plays a role in shaping engine intakes, which must be designed to efficiently channel airflow into the engine and propeller blades, which generate thrust through their motion through the air.

Given the importance of the airfoil in determining an aircraft's aerodynamic properties, selecting airfoils are crucial in aircraft design. This is especially true for wings, which are the primary source of lift and drag forces on an aircraft. Accurate prediction of wing aerodynamics is essential for designing aircraft that are capable of stable, efficient flight. The design of other airfoil-shaped components, such as control surfaces and engine intakes, is also essential for optimizing an aircraft's performance. Overall, the airfoil plays a vital role in the design and performance of aircraft and is an important area of study in the field of aerodynamics.

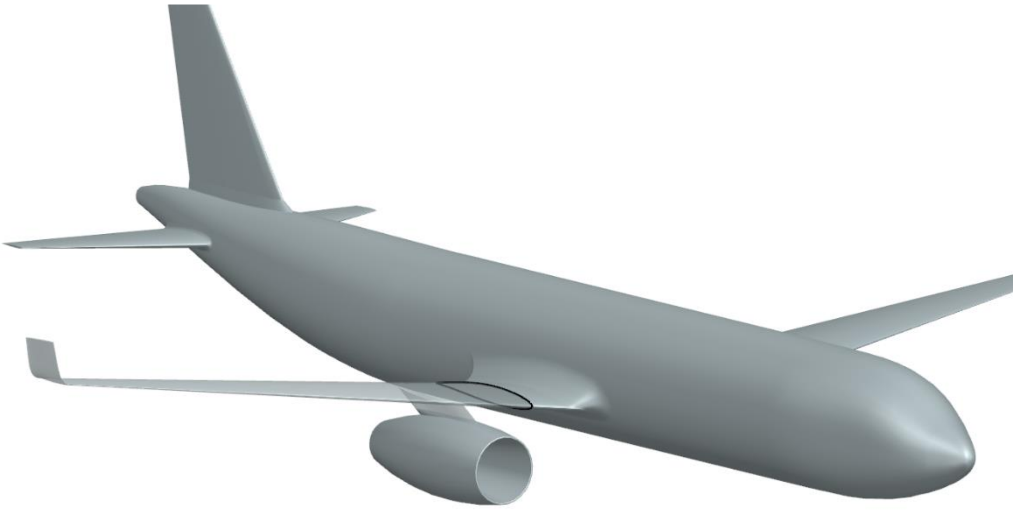

Figure 1.1 Wing cross-section of a generic airliner

The preliminary design phase of an aircraft is an important stage in the aircraft design process, where initial concepts are developed and refined into a more detailed design [2]. During this phase, a wide range of design considerations are taken into account, including the aircraft's mission, performance requirements, structural layout, weight and balance, systems integration, and manufacturing considerations.

During the preliminary design phase, the aircraft's overall configuration is established, and the general layout of its major components, such as the wings, fuselage, tail, and engines, is determined. The designers consider the aircraft's aerodynamic characteristics, including lift, drag, and stability, as well as its propulsion system and power requirements.

This phase also includes a trade-off study where different configurations, materials, and systems are evaluated to obtain the best balance of performance, cost, and risk. Aircraft specifications are also defined at this phase, including weight, size, range, speed, and other performance metrics. This design phase is iterative and requires aerodynamic calculations for many different design alternatives.

Typically, wind tunnel tests or CFD analyses are used in later design phases to obtain aerodynamic data of the design. However, these methods are not feasible for the preliminary design phase since many conceptual designs should be tested, and hence it would be costly in time and money. Thus, the need for a tool that can provide aerodynamic information rapidly arises in the preliminary design phase.

Low-accuracy methods with rapid predictions exist in the literature, but they mainly fall short on providing reliable solutions for fully turbulent and transonic flows. A review of these methods is presented in Chapter 4. These methods offer successful enough results in subsonic and low speeds, but their accuracy deteriorate at higher speeds. This shortcoming is especially prominent in modern aircraft design. For example, typical airliners cruise at transonic speeds where these tools do not provide accurate results. A tool that can provide rapid predictions in the transonic and fully turbulent regime is needed.

. The shape of an airfoil, which includes its camber, thickness, and curvature, directly affects the flow of air over the surface of the airfoil and the resulting forces and moments (e.g., drag, lift, and pitching moment). In addition, the cruise speed of transonic airliners which resides near drag-divergence, is also dictated largely by airfoil shape. Moreover, small variations in airfoil shape may result in the transonic behavior of airfoils. Therefore, it is crucial to describe airfoil shapes accurately when evaluating their aerodynamic performance. A detailed investigation of the relationship between airfoil shape and performance will be provided in Chapter 2.

1.1 Aim and scope of this study

In this study, we aimed to develop an accurate enough tool that can provide predictions of aerodynamic coefficients (lift, drag, and drag-divergence Mach number) in transonic and fully turbulent flow. The flow conditions should include post-stall conditions, too. Also the tool should provide estimations rapidly, and should be coupled with a robust and complete shape descriptor. Our scope is to use this tool in predictions of supercritical airfoils primarily to be used, but not limited to, for designing commercial airliners. This tool can generate the many required aerodynamic evaluations of the iterated aircraft geometry in preliminary design. Also, by predicting the drag-divergence Mach number, the need for expensive RANS calculations will be eliminated for the preliminary design phase. Provided estimations will pick a good starting point for the detailed design phase. Such a tool is not available in existing research nor commercially. To achieve this goal, we need a proper shape descriptor, a prediction methodology, and an accurate and consistent aerodynamic database to develop this tool.

1.2 Contents of this study

An investigation of airfoil's aerodynamic performance and how it relates to flow conditions and shape is presented in Chapter 2. In Chapter 3, a review of standard methods for parametrizing the airfoil shape is given. Chapter 4 contains reviews of tools, either presented in literature or commercial, that can provide rapid predictions of airfoil performance and their advantages/disadvantages. Chapter 5 contains studies regarding neural network construction on already existing data. This chapter provides proof of concept in using multi-layer perceptron to create our prediction tool. Chapter 6 contains information about how the data used in this study was gathered and studies regarding the data validation. Finally, Chapter 7 contains the final neural network model that can provide predictions in transonic and turbulent regimes.

CHAPTER 2

REQUIREMENTS FOR ACCURATE AIRFOIL PERFORMANCE ESTIMATIONS

2.1 Flow and shape parameters that affect airfoil performance

The aerodynamic performance of an airfoil is primarily determined by three flow parameters: the angle of attack, the Mach number, and the Reynolds number. The angle of attack, as seen in Figure 2.1, is the angle between the chord line and the relative wind. Mach number is the dimensionless number that describes the compressibility of a flow, which is described as the ratio of relative wind speed to the speed of sound. The Reynolds number represents the ratio of inertial and viscous forces.

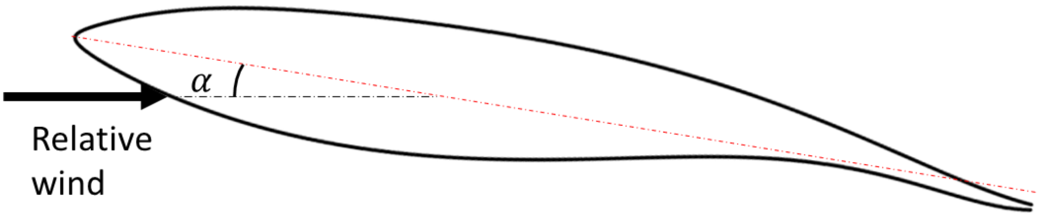

Figure 2.1 Angle of attack definition

Drag and lift are two of the most critical aerodynamic forces on an airfoil. Drag force is the resultant force aligned with the relative wind and can be considered the wind resistance against a moving aircraft. Lift force is the component of the resultant force perpendicular to the relative wind. The dimensionless drag and lift coefficients are defined by dividing the respective force by the dynamic pressure and reference area.

$$C_d = \frac{D}{0.5\rho V^2 S_{ref}} \qquad 2.1$$

$$C_l = \frac{L}{0.5\rho V^2 S_{ref}} \qquad 2.2$$

The airfoil produces lift by moving the fluid around it so that the lower side has more pressure than the upper side. This difference in pressure results in a force that lifts the airfoil perpendicular to the wind. A typical pressure distribution for an airfoil can be seen in Figure 2.2.

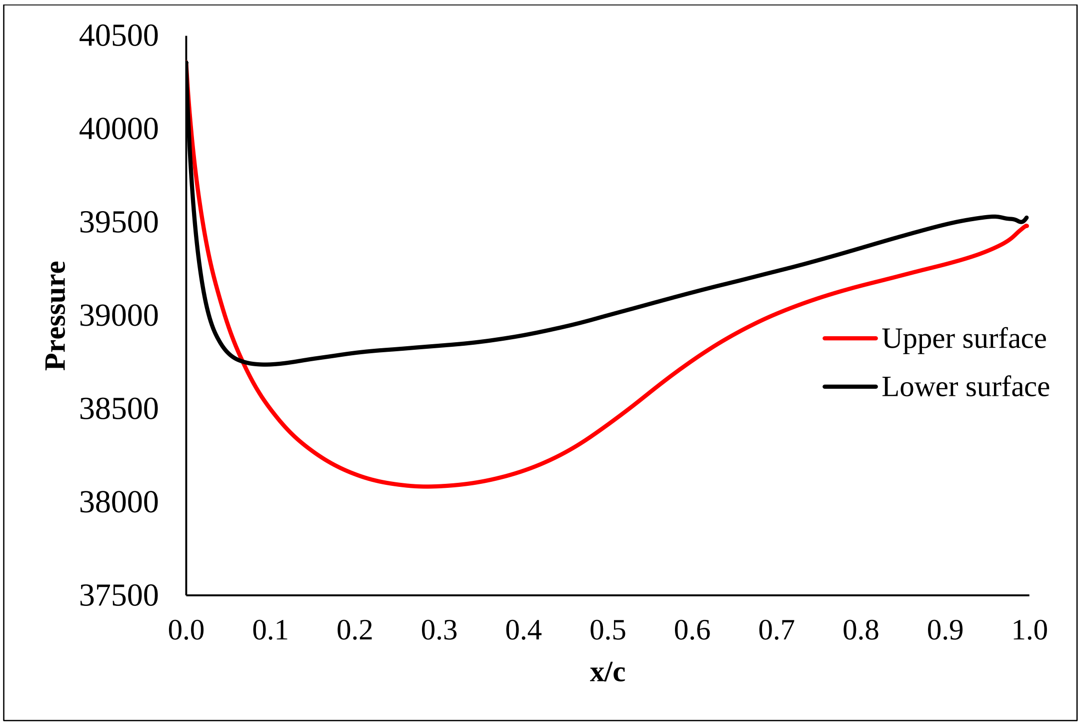

Figure 2.2 A typical pressure distribution over the airfoil

2.2 Common Airfoil Shape Representations

The airfoil is a 2D curve that can be represented in various methods. A couple of formats in literature have been used to represent them. The Selig format is a file format used to represent the geometry of an airfoil as an ordered point list. The format was developed by Michael Selig, a researcher at the University of Illinois at Urbana-Champaign, and is commonly adopted by the aerodynamics community [3]

In the Selig format, the coordinates of an airfoil's upper and lower surfaces are represented by a set point with (x, y) coordinates. The x coordinate represents the position along the airfoil's chord, and the y coordinate represents the distance from the chord. The coordinates are typically represented in a text file, with one $x\ and\ y$ pair per line. The file typically starts with the upper surface trailing edge coordinate and then lists the coordinates for the upper surface of the airfoil, followed by the coordinates for the lower surface of the airfoil, and finishes with the lower surface trailing edge coordinates. An excerpt of a Selig format airfoil can be seen in Table 2.1.

The Selig format is widely used due to its simple use and can be easily read by most airfoil analysis software. The format is also supported by many airfoil design tools, which makes it easy to import airfoil geometry into these tools. Also, the format allows easy modification or the generation of new airfoils by using the coordinates.

Table 2.1 An example Selig format airfoil

bac1.dat	
1	0.00106
0.99	0.00283
0.98	0.00461
0.97	0.00639
…	
0.0005	0.00242
0	0
…	
0.99	-0.0003
1	-0.00096

Other formats are used to store airfoil coordinates, such as the NACA format and Lednicer format. Selig format, however, is widely used in the research community and industry, and large data sets are freely available, like the UIUC database [4].

While the Selig format is widely used in the industry, the number of segments given in format files for each airfoil varies greatly. Moreover, the exclusion of curvature information leads to incomplete airfoil shape definition. For example, we may need more than 1000 points in both the upper and lower surfaces of the airfoil in CFD, which is rarely the resolution presented in the airfoil databases. An accurate, reusable and extensible shape parametrization method is required that can extract principal information from the airfoil coordinates and accurately represent the airfoil as a continuous function. A detailed survey of common parametrization methods is given in Chapter 3.

2.3 Transonic Flow around the Airfoils

The Mach number, the ratio of an aircraft's speed to the speed of sound, is a crucial factor in the design and operation of transonic aircraft such as commercial airliners. As the Mach number approaches unity, wave drag, a form of drag caused by the formation of shock waves, begins to increase significantly. The Drag Divergence Mach Number (Ma_{DD}) is particularly relevant in the transonic flow regime. This number denotes the Mach number at which the drag coefficient starts to increase rapidly with increasing flow speed. Commercial airliners typically design their wings to cruise near this Mach number to optimize performance and fuel efficiency. Speeds above $\mathrm{Ma}_{\mathrm{DD}}$ require a significant increase in thrust, which is not desirable. The drag rise after the $\mathrm{Ma}_{\mathrm{DD}}$ for a typical transonic airfoil can be seen in Figure 2.3 marked with a red circle.

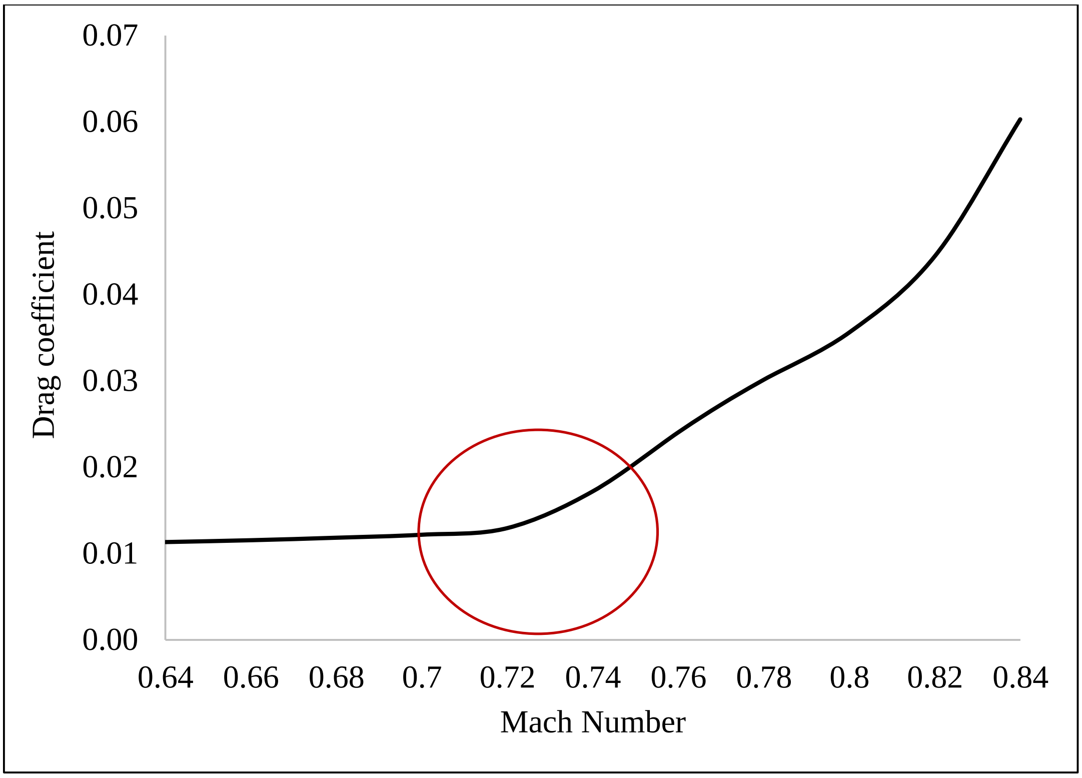

Figure 2.3 Drag vs. Mach graph of SC2-0714 airfoil

2.3.1 Drag Divergence Definition

One of the earliest definitions of drag divergence was given in a NASA Technical Note [5] as "Mach number at which slope of the curve of drag coefficient versus Mach number attains a value of 0.10". Components of the drag coefficient are usually categorized into two: the base drag (C_{D_0}) and the induced drag (due to lift) (KC_L^2), as represented in Equation 2.3.

$$C_d = C_{d_0} + KC_l^2 \qquad 2.3$$

The increase in wave drag due to increasing Mach number builds up an increase in the base drag, C_{D_0} component, which is independent of lift, is called drag rise. To isolate the effect of Mach number, the definition of drag divergence Mach number used in this thesis is given in 2.4.

$$\left.\frac{\mathrm{dC_d}}{\mathrm{dMa}}\right|_{\mathrm{C_l=constant}} = 0.05 \; @ \; \mathrm{Ma_{DD}} \qquad 2.4$$

Equation 2.4 reveals that drag divergence Mach Number, $\mathrm{Ma_{DD}}$, is different at each lift coefficient. CFD results for the airfoils consist of discrete points, meaning they are not analytical functions, so numerical methods are required to calculate $\mathrm{Ma_{DD}}$ with the help of numerical derivatives.

With the use of numerical methods, the angle of attack that corresponds to the constant lift coefficient is found for each Mach number. The drag coefficient corresponding to that angle of attack is then calculated. After this is repeated for each Mach number, the derivative in Equation 2.4 can be calculated. One should keep in mind that applying this method to CFD results can introduce numerical errors.

2.4 Physical parameters that affect airfoil performance

The airfoil shape is one of the most critical factors that define aerodynamic performance. The physical parameters like maximum thickness, maximum camber, leading-edge radius, boat-tail angle, and others have various effects on the aerodynamic performance of airfoils. For example, in Raymer's textbook for aircraft design [6], the maximum thickness percentage has been shown to affect drag and maximum lift coefficients. For instance, the maximum thickness for airfoils from NACA 4-digit family is reported [6] to increase the drag coefficient, as seen in Figure 2.4. The maximum thickness also affects the maximum lift coefficient, and this effect is different for every airfoil family, as seen in Figure 2.5.

In a study published in 2015 by Birajdar and Kale [7], the authors investigated the effects of leading-edge radius. By taking NACA4412 as the baseline airfoil, they tried different airfoils by tweaking the leading edge and examining the results. It was shown that the changes in leading-edge radius change the L/D drastically.

Another study investigating the effect of the airfoil shape on rotorcraft performance [8] showed that parameters that define the shape, like camber, leading-edge radius, thickness, and boat tail angle, significantly affect the overall performance of the rotor. The shape must be adequately defined to evaluate the airfoil's performance correctly.

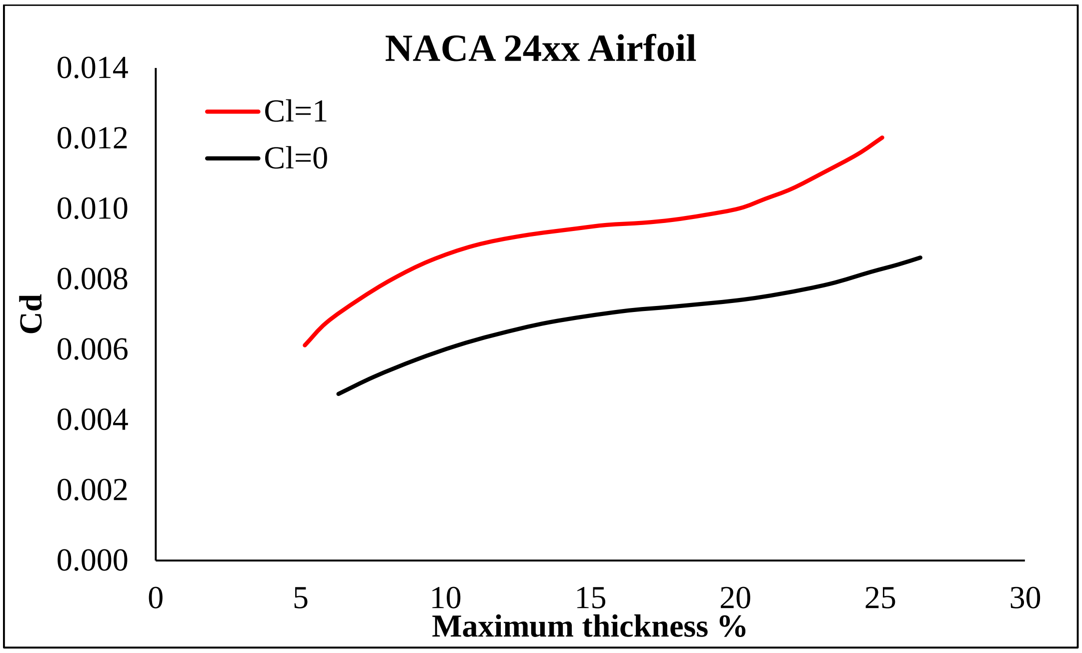

Figure 2.4 Effect of maximum thickness on drag, taken from [6]

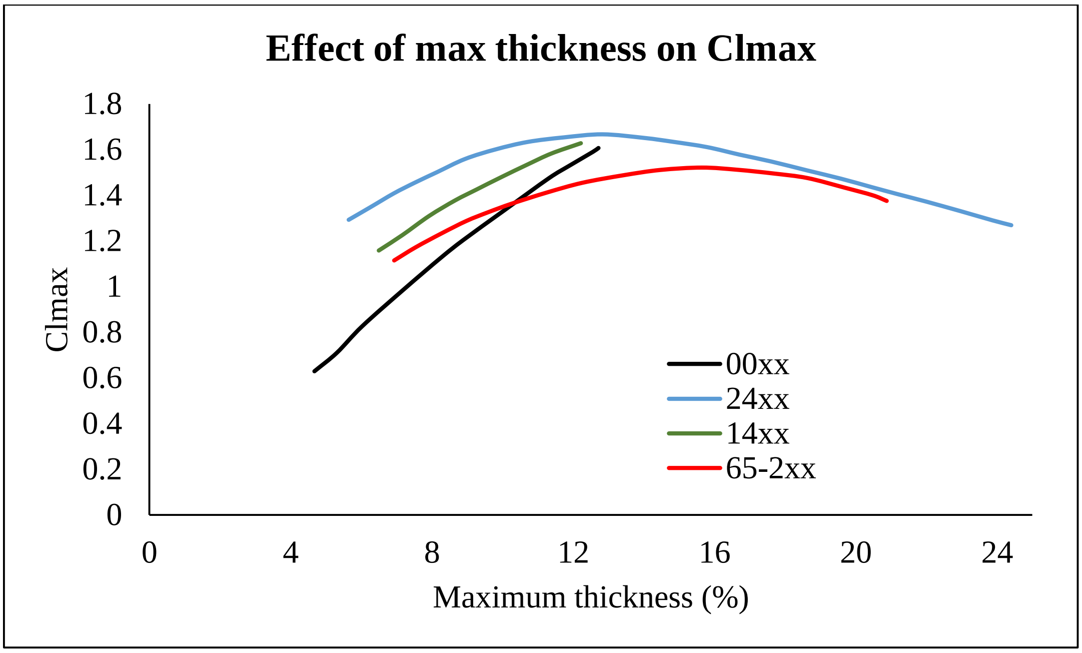

Figure 2.5 Effect of maximum thickness on the maximum lift, taken from [6]

CHAPTER 3

COMMON SHAPE PARAMETRIZATION METHODS

Most of the airfoils in the literature are presented as 2D coordinates. While some of these airfoils can have as many as 200 points, some have as few as 40 points. Representing these airfoils as analytic curves instead of a discrete set of points is needed in most applications that use airfoils, e.g., CAD software, CFD applications, machine learning applications, etc.

Many methods are present in the literature that tackles this problem. These methods have their strengths and weakness for each application which we will investigate in this chapter. The accuracy of these methods will be judged by their absolute errors, e.g., the vertical distance between legacy points and the parametrized curve and not the orthogonal distance.

3.1 Interpolation

Using interpolating curves to describe an airfoil is one of the first methods that come to mind because of its simplicity. However, interpolation is not straightforward for some applications. For example, in the flows above or near the sound of speed, the changes in slope and curvature play a significant role. A high-order interpolation is required to reach acceptable accuracy. However, high-order interpolation will introduce fluctuations in curvature graphs, which is not desired in transonic and supersonic flows. These fluctuations can be seen in Figure 3.1. Sudden changes in curvature will result in various expansion and shock waves across the airfoil surface, which will diminish the performance of CFD and wind tunnel tests. These waves can be seen in Figure 3.2, which is the CFD result of an interpolated airfoil in transonic flow.

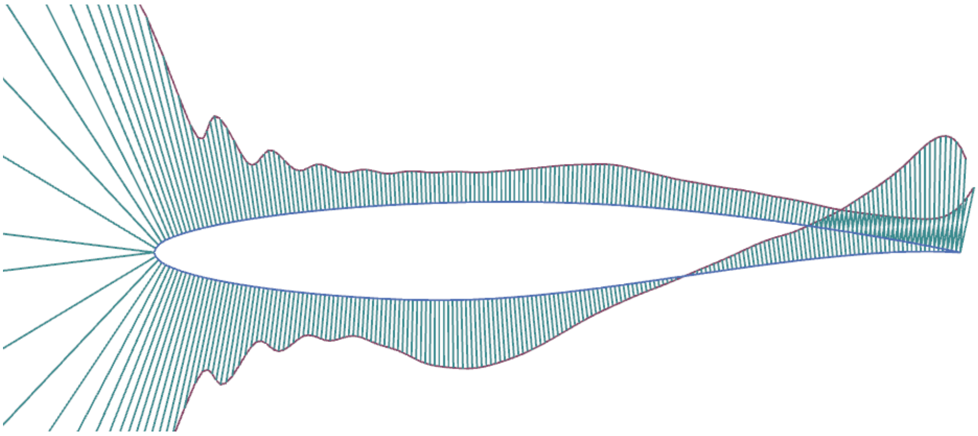

Figure 3.1 Curvature graph of an interpolated airfoil

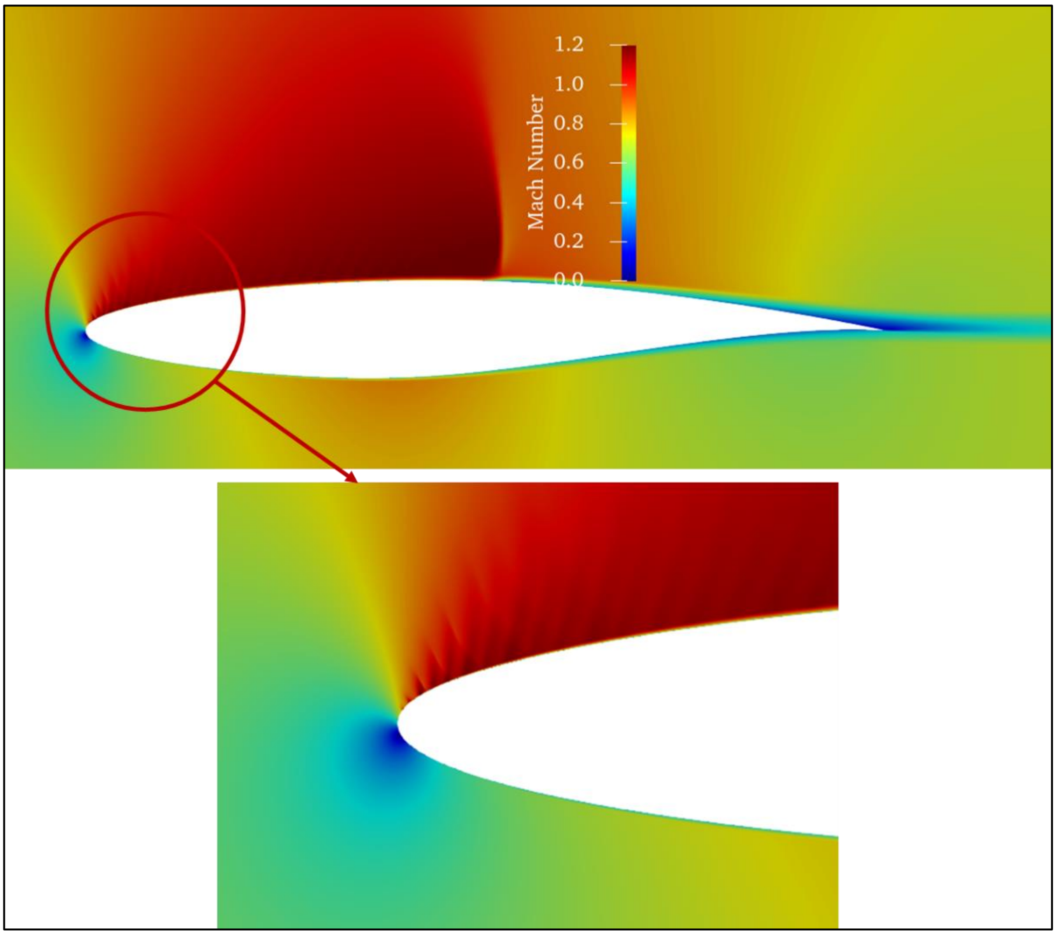

Figure 3.2 Series of expansion and compression waves across an interpolated airfoil surface

Another interpolation method is using splines. While using splines solve the fluctuation problem with polynomial interpolation, they require a lot of parameters to represent an airfoil within acceptable accuracy. For example, to represent the upper surface of RAE2822, a supercritical airfoil, we used 20 degrees of freedom. This trial yielded a maximum error of 8.358e-04, which is not ideal. This shows that the spline interpolation is an inferior shape descriptor for machine learning because the parameters are too many compared to the methods described in the following sections.

3.2 Beziér Curves

Beziér curves are readily available in most CAD packages, and it is easy to control the shape by manipulating the control points. They are constructed by multiplying Bernstein polynomials with each corresponding control point [9]. A Beziér curve of order n can be represented as in Equation 3.1.

$$\mathbf{B}(\mathrm{t}) = \sum_{\mathrm{i}=0}^{\mathrm{n}} \binom{\mathrm{n}}{\mathrm{i}} (1-\mathrm{t})^{\mathrm{n}-\mathrm{i}} \mathrm{t}^{\mathrm{i}} \mathbf{P}_{\mathrm{i}} \qquad 3.1$$

Where, $\binom{\mathrm{n}}{\mathrm{i}}$ is the binomial coefficient, $\mathbf{P}_{\mathrm{i}}$ vector is the control point vector (x_i, y_i), and $\binom{\mathrm{n}}{\mathrm{i}} (1-\mathrm{t})^{\mathrm{n}-\mathrm{i}} \mathrm{t}^{\mathrm{i}}$ term is the Bernstein polynomial [10]. These multiplications result in $\boldsymbol{x(t)}$ and $\boldsymbol{y(t)}$ curves ($\boldsymbol{B}(t) = [\boldsymbol{x}(t) \quad \boldsymbol{y}(t)]$) dependent on the parameter $\mathrm{t} \in [0,1]$. However, the relationship between the control points and the shape is not easily realizable. The Bernstein polynomials are multiplied with both the x and y coordinate of the control point, which couples the contribution of x_i and y_i parameters. This coupling increases the difficulty of the underlying relationship between the aerodynamic coefficients and the input parameters. Moreover, the contribution of a single control point is dependent on other control points, and they are not independent of each other. Finally, representing legacy airfoils with Bezier curves requires significant effort since the **t** vector is nonlinear. A sample legacy airfoil transformation is presented in Figure 3.3.

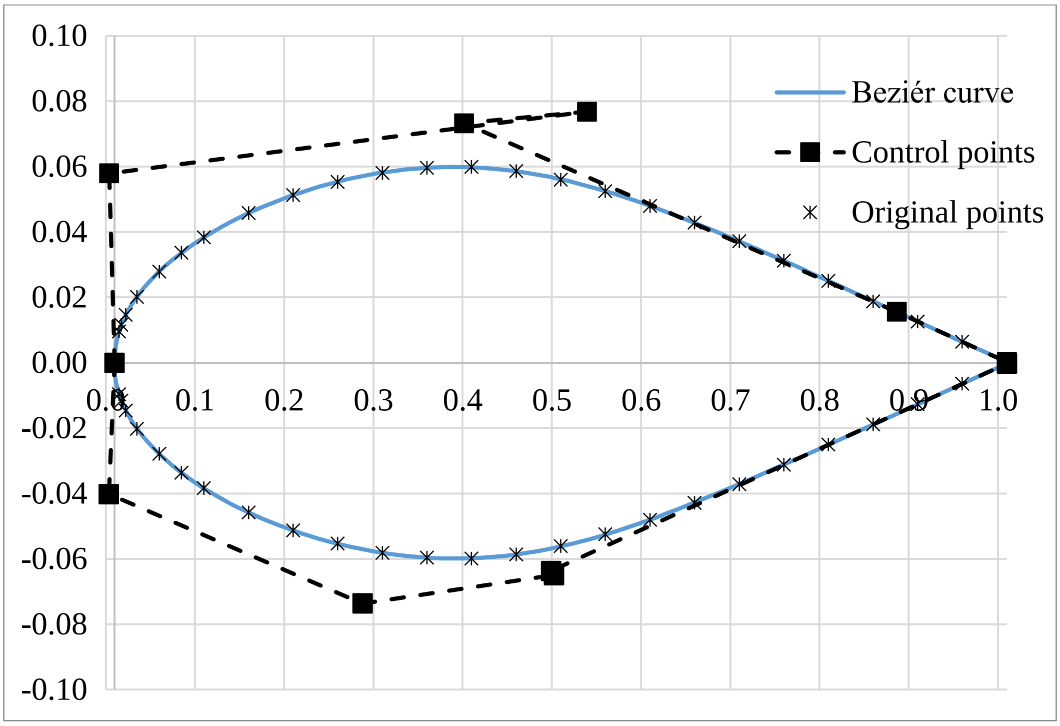

Figure 3.3 Beziér curve representation of NACA 64-012a airfoil

3.3 Class-Shape Function Transformation

Class-Shape function transformation proposed by Kulfan [11] overcomes most of the problems associated with polynomial curves. This method, given in Equation 3.2, separates the generating function into two parts: the class function and the shape function. For airfoils, generally, the class function of $(1-x)\sqrt{x}$ is used, which has two distinct features: it is round at the leading edge (at $x = 0$, thanks to the $\sqrt{x}$ part) and vanishes at the trailing edge (at $x = 1$). This class function is multiplied by the shape function, which consists of a series of linearly independent Bernstein polynomials. The author presents the idea that the shape function is dependent on physical parameters, e.g., leading-edge radius, maximum thickness, etc. Notice some airfoil performance features like the boat-tail angle and leading-edge radius are apparent in the shape functions. How the shape function affects these parameters is highlighted in Figure 3.4 presented in the original study.

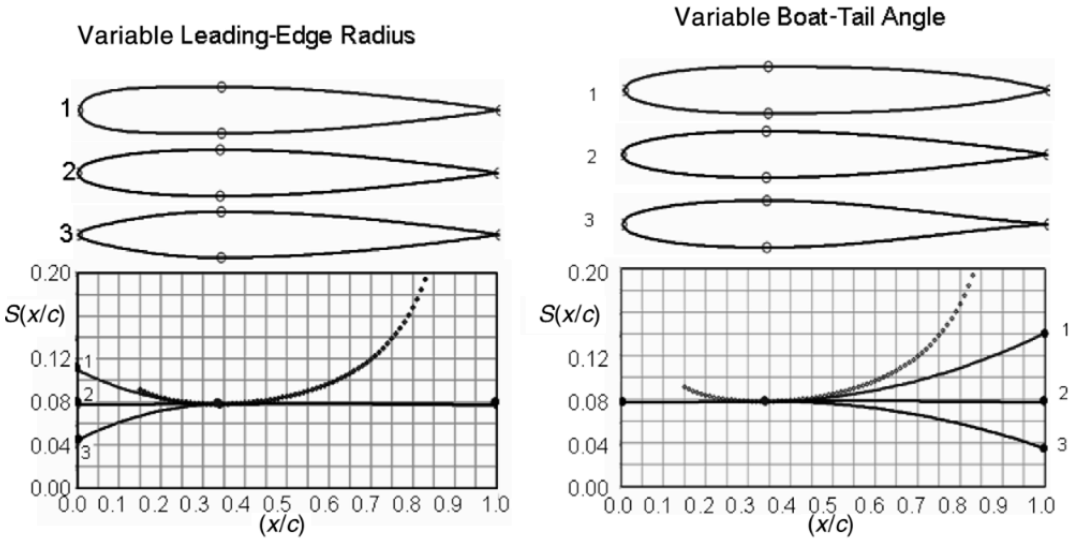

Figure 3.4 Fig.2 of reference

$$y(x, \mathbf{v}, z_{TE}) = \underbrace{x^{N2}(1-x)^{N1}}_{\text{Class function}} \underbrace{\sum_{r=0}^{n_{order}} v_r \; C\binom{n}{r} x^r (1-x)^{n-r}}_{\text{Shape functions}} \underbrace{+z_{Te}x}_{\text{Trailing edge term}} \qquad 3.2$$

Each of the Bernstein polynomials constructing the shape function is multiplied with the corresponding CST coefficient and added together. Since the Bernstein polynomials are linearly independent, we can see that each coefficient's contribution to the parametrization is independent of each other as well.

Transformation of legacy airfoils is also easily implemented using an ordinary least squares method which will be covered in the next section.

3.3.1 Legacy Airfoil Representation with CST

To transform an airfoil given in Selig, we would first need to represent lower and upper surfaces separately, so we split the airfoil data. Equation 3.2 is valid only for the upper or lower surface of the airfoil. This representation would greatly reduce the data dimension and would enable interpolation between intermediate points in the Selig format.

$$\mathbf{x}_u = [1 \quad 0.99 \quad 0.98 \quad \dots \quad 0.01 \quad 0.005 \quad 0.002 \quad 0]^T \tag{3.3}$$

$$\mathbf{z}_u + \mathrm{z}_{\mathrm{TE}}\mathbf{x}_u = [-0.0095 \quad -0.0063 \quad -0.0032 \quad \dots \quad 0.01658 \quad 0.01077 \quad 0]^T \tag{3.4}$$

where $z_{TE} = -0.0095$ in this case. The vector given as $\mathbf{z}_{\mathrm{u}} + \mathrm{z}_{\mathrm{TE}}\mathbf{x}_{\mathrm{u}}$ is the z coordinate of the airfoil given in legacy format. We subtract $\mathrm{z}_{\mathrm{TE}}\mathbf{x}_{\mathrm{u}}$ from the z coordinates to isolate $\mathbf{z_u}$ vector in Equation 3.4. If we can represent this transformation in the form of $\mathbf{B}^{\mathrm{u}}\mathbf{v}^{\mathrm{u}} = \mathbf{z}_{\mathrm{u}}$, the coefficient vector, $\mathbf{v^u}$, can be calculated using Moore–Penrose inverse with the relation in Equation 3.5:

$$\mathbf{v}^{\mathrm{u}} = \left(\mathbf{B^u}^{\mathbf{T}}\mathbf{B^u}\right)^{-1}\mathbf{B^u}^{\mathbf{T}}\mathbf{z^u} \tag{3.5}$$

For example, the **B** matrix for a 5th-order transformation can be constructed with the following relations:

$$\begin{aligned}
\mathrm{F}_{\mathrm{i}_0} &= \sqrt{\mathrm{x_i}}(1-\mathrm{x_i})\mathrm{C}\binom{5}{0}\mathrm{x}^0(1-\mathrm{x})^5 \\
\mathrm{F}_{\mathrm{i}_1} &= \sqrt{\mathrm{x_i}}(1-\mathrm{x_i})\mathrm{C}\binom{5}{1}\mathrm{x}^1(1-\mathrm{x})^4 \\
\mathrm{F}_{\mathrm{i}_2} &= \sqrt{\mathrm{x_i}}(1-\mathrm{x_i})\mathrm{C}\binom{5}{2}\mathrm{x}^2(1-\mathrm{x})^3 \\
\mathrm{F}_{\mathrm{i}_3} &= \sqrt{\mathrm{x_i}}(1-\mathrm{x_i})\mathrm{C}\binom{5}{3}\mathrm{x}^3(1-\mathrm{x})^2 \\
\mathrm{F}_{\mathrm{i}_4} &= \sqrt{\mathrm{x_i}}(1-\mathrm{x_i})\mathrm{C}\binom{5}{4}\mathrm{x}^4(1-\mathrm{x})^1 \\
\mathrm{F}_{\mathrm{i}_5} &= \sqrt{\mathrm{x_i}}(1-\mathrm{x_i})\mathrm{C}\binom{5}{5}\mathrm{x}^5(1-\mathrm{x})^0 \\
\mathrm{F}_{\mathrm{i}_6} &= \mathrm{x_i}\sqrt{1-\mathrm{x_i}}(1-\mathrm{x})^5
\end{aligned} \tag{3.6}$$

where $i \in 1,2,3, \dots, n$ denotes each x coordinate of the legacy airfoil.

$$\mathbf{B} = \begin{bmatrix} F_{1_0} & F_{2_0} & \dots & F_{n_0} \\ F_{1_1} & F_{2_1} & \dots & F_{n_1} \\ F_{1_2} & F_{2_2} & \dots & F_{n_2} \\ F_{1_3} & F_{2_3} & \dots & F_{n_3} \\ F_{1_4} & F_{2_4} & \dots & F_{n_4} \\ F_{1_5} & F_{2_5} & \dots & F_{n_5} \\ F_{1_6} & F_{2_6} & \dots & F_{n_6} \end{bmatrix}_{6\text{ x n}} \qquad 3.7$$

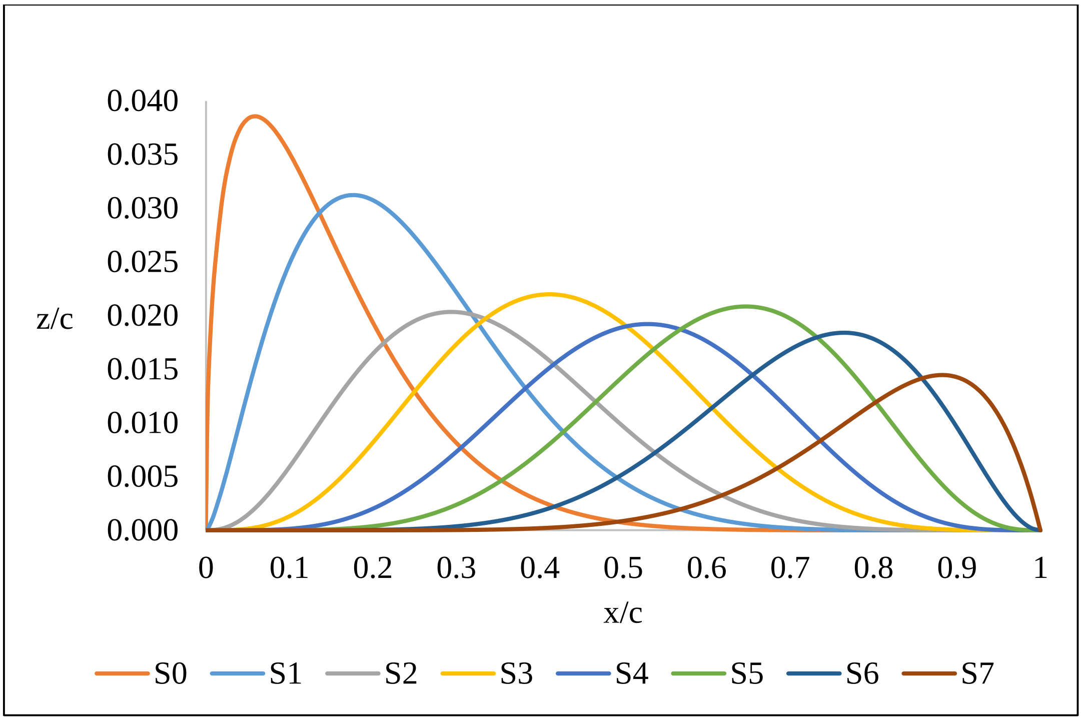

Figure 3.5 Contribution of each shape function to the transformation of SC2-0714 airfoil

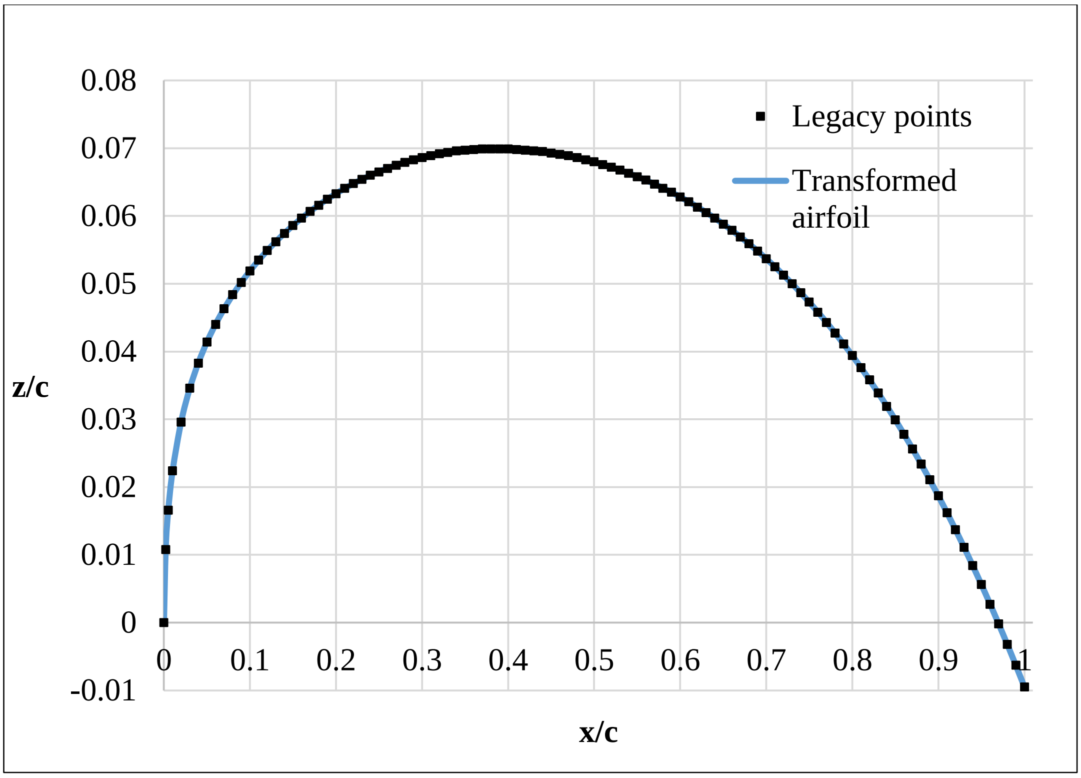

Figure 3.6 7th order transformation of the SC2-0714 legacy airfoil

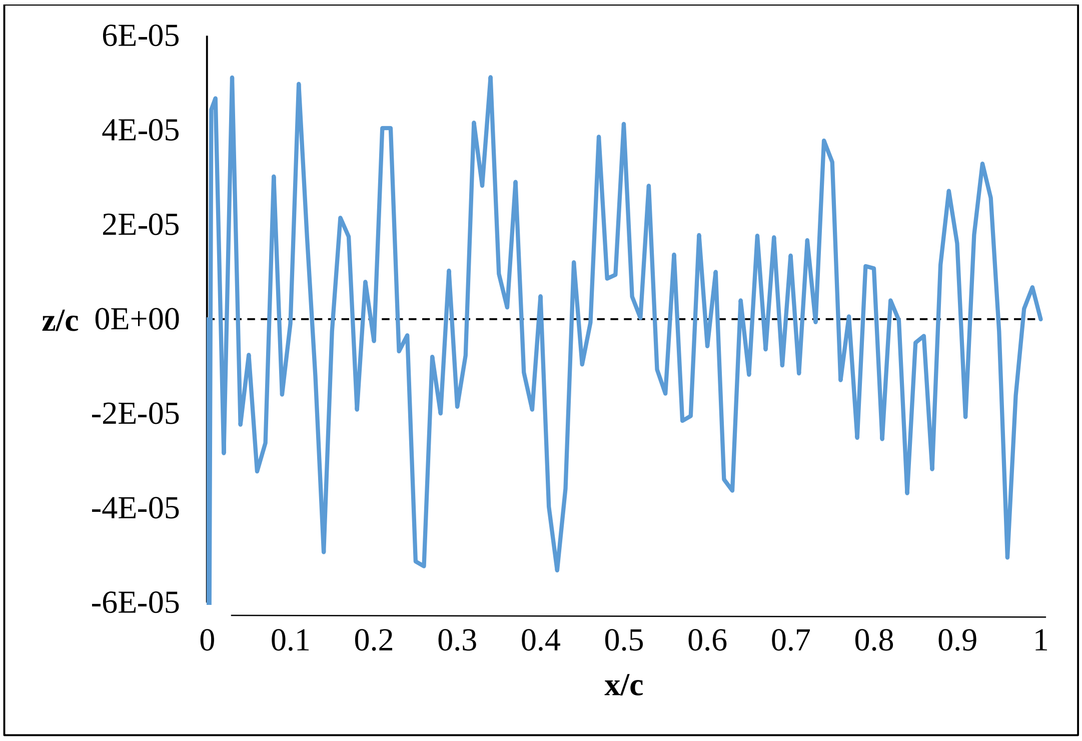

Figure 3.7 Error plot for CST of SC2-0714 airfoil

Figure 3.6 and Figure 3.7 show an example CST of SC2-0714 legacy airfoil. Curves in Figure 3.5 correspond to each shape function multiplied by the class function. The sum of each of these components completes the transformation, as seen in Figure 3.6. The transformation is completed by solving Equation 3.5. The original airfoil representation in Selig format consists of 205 points. In the end, we reduced the airfoil to 20 parameters with a maximum error of 0.06 mm for a meter-long chord. It is noted that this error is below typical tolerances for wind tunnel models, which means that this transformation describes this airfoil precisely. The airfoil in question has a complicated shape to parametrize, especially towards the leading edge, which requires increased complexity in most methods to represent correctly. For example, if Beziér curves are used, we would need a 6th-order or higher curve to represent RAE2822 within the same error band, which would result in 28 parameters. That would result in more predictors than CST, which might diminish model performance. To obtain the same error band, we would need the following:

◦	Spline interpolation:	40 degrees of freedom
◦	Beziér curves:	26 control points
◦	**CST method:**	**20 points**

A neural network model constructed with both CST and Beziér curves will be compared in Chapter 5.

CHAPTER 4

RAPID PREDICTION TOOLS IN AIRFOIL PREDICTIONS

This chapter gives brief reviews of rapid prediction tools that exist in literature or that are commercially available. The advantages and disadvantages of each tool and their application to transonic and fully-turbulent regimes are also discussed. Moreover, existing studies in the literature regarding neural networks and rapid predictions are discussed in 4.2. A summary of these studies is presented at the end of the chapter.

4.1 Low-Fidelity Commercial Tools

4.1.1 DATCOM

USAF DATCOM is a software released in 1976 to help with aircraft design, specifically for stability and control [12]. It can generate static coefficients for 3D geometries, so it is possible to evaluate a complete aircraft. However, in the scope of this paper, we will focus on the airfoil section module. This software accepts airfoil coordinates, camber and thickness distribution, or common shapes as inputs. After the airfoil shape is described, it uses inviscid formulations to calculate aerodynamic coefficients and later applies a viscous correction to the lift coefficient.

While this program can generate results rapidly and accurately in some cases, it falls short in many others. It is noted in the DATCOM description page [13] that the program will have difficulties when the Mach number is near drag divergence or when the maximum camber exceeds 6%. The user will not get accurate results for high-camber airfoils. It is also noted that airfoils exceeding %12 thickness or supercritical airfoils will have limited accuracy. The accuracy of DATCOM solutions also diminishes with increasing the angle of attack.

4.1.2 XFOIL

XFOIL is a program developed at MIT to be used to design and analyze subsonic airfoils. It is based on a high-order panel formulation coupled with various corrections [14]. XFOIL offers airfoil solutions for variable Reynolds numbers and Mach numbers, an inverse design feature, and an airfoil redesign feature. It provides accurate results for low Reynolds numbers and inviscid flows but is only available for partially turbulent and subsonic flows. The flow regime in the scope of this study is fully turbulent and transonic, in which XFOIL has trouble with convergence and provides questionable results.

4.1.3 Vortex Lattice Method

Vortex Lattice Method [15] is a low-order numerical method used to obtain computational fluid dynamics. It uses the assumption that the flow is inviscid, incompressible, and irrotational. For thin wings below the stall angle of attack, it can provide an idea about the aircraft's performance. The stability and control derivatives can be derived from the aerodynamic coefficients, which can provide some ideas in the early stages of design. However, this method falls short of providing meaningful calculations for thicker wings and post-stall conditions. Since supercritical airfoils are usually thicker and highly cambered, this method is not feasible in this regime.

4.2 Neural Network Methods

Numerous studies have been conducted to generate rapid solutions for airfoils. These studies consist of empirical relations, low-order approximations of fluid equations such as panel methods and inviscid flows, surrogate models such as neural networks or decision trees, and many more. Many methods for airfoil performance predictions have been proposed on the topic of neural networks, e.g., multi-layer perceptrons, convolutional neural networks, and others. Most of these methods have advantages and disadvantages and limits to their applications. The following sections present a detailed analysis of some of these studies. A summary of the reviews can be seen in Table 4.1.

4.2.1 Artificial Neural Networks

Artificial Neural Networks (ANNs) are a type of machine learning model that are inspired by the structure and function of biological neural networks. They consist of layers of interconnected “neurons” that process and transmit information. They have been used in numerous areas, such as image recognition, language processing, and many more. Their ability to learn the nonlinear nature of data has made them very popular for providing rapid predictions. However, a way to represent airfoils, an airfoil parametrization method, is needed to provide input to the ANN algorithm instead of discrete airfoil points, like in Selig format.

Suresh et al. [16] employed recurrent neural networks to estimate the dynamic stall effect in rotor blades on various pitch angles. This study was conducted to predict a single airfoil and is one of the earlier applications of neural networks for airfoil prediction. In the end, the authors reported an R^2 score of around 0.99 for various pitch angles.

Santos et al. [17] parametrized the airfoils using Sobieczky's generic airfoils method. They employed XFOIL to generate results in subsonic conditions and used an inviscid CFD code in the transonic region. The inputs to Sobieczky's method, angle of attack, and Mach number were chosen as the input layer to a Multi-Layer Perceptron algorithm. The machine learning model was trained to learn the airfoils' drag and lift coefficients. One thing to note is that the data in this study does not include the post-stall region, nor does it include the near-drag-divergence region. In the end, the authors achieved an R^2 score of 0.991 for the lift coefficient, which can be considered successful.

In another study, Balla et al. [18] employed a proper orthogonal decomposition (POD) to reduce pressure coefficient graphs to POD coefficients. They used non-uniform rational B-Splines (NURBS) to parametrize the airfoils, resulting in 25 independent geometric parameters. Then, they trained a model that can construct the relation between the POD coefficients and the NURBS parameters. Flow solutions for aerodynamic database construction were obtained with CFD, using inviscid models in a wide Mach number range (0.3-0.9). After the model construction was completed, the generated pressure coefficient graphs were used to calculate aerodynamic coefficients. While this study presented excellent results in predicting lift coefficient, the flow conditions or shape parameters were limited for each model.

Xiaosong et al. [19] employed Multi-Layer Perceptron (MLPs) to generate a surrogate model before an optimization problem. They used CFD to generate solutions with a modified B-Spline parametrization and obtained an RMSE of $2.77 * 10^{-4}$ for the drag coefficient and $12.90 * 10^{-4}$ for lift coefficient. These errors are considerably small but only up to 0.7 Mach number.

4.2.2 Convolutional Neural Networks

Convolutional networks have also been used to recognize airfoil shapes and train for aerodynamic coefficients. CNN is handy for image recognition applications. The method works best when the input is highly nonlinear and has a spatial correlation. CNN methods mainly utilize image-like grid data structures as input variables and can be trained for various outputs. While these networks can have a broader scope, such that they can predict the flow field around the airfoil, pixelating the airfoil decreases details in the airfoil shape on a reasonable resolution.

In the study by Jin et al. [20], they employed CNN to study flow past a cylinder at low Reynolds numbers. They used an incompressible Navier-Stokes solver to generate data. They represented the velocity fields as grid-like images to the CNN algorithm and trained them to minimize velocity errors. In the end, they obtained scores above 0.995 for both u^* and v^*.

The method proposed by Chen et al. [21] employs Hicks-Henne bump functions on top of the NACA0012 geometry to parametrize the airfoil shape. They used convolution to incorporate the angle of attack and Mach number into the airfoil image to be used as input to the CNN model. The aerodynamic database used in the study was defined between 0.1-0.6 Mach number and 2-15 degrees of angle of attack. Their study results in an RMSE of 0.0273 for the lift coefficient. While the error they achieved is significantly low considering the high AoA range, it does not include the transonic regime.

In the study by Duru et al. [22] [23], the authors constructed a CNN model to predict the flow field around airfoils. Their dataset consists of a wide angle of attack range (-10° to 20°) at a transonic Mach number of 0.7 with various kinds of airfoil shapes and families. This data contains shock-induced separation cases and post-stall cases. The airfoil shape was given to the model as a distance map, and the output is the flow variables around the airfoil. This study's scope is broader than predicting lift and drag coefficient and can provide helpful information regarding lift and drag. The pressure prediction around the airfoil was used to calculate the lift and drag coefficient by integrating it around the airfoil. The lift coefficient and drag polar presented for NACA 66(3)-218 airfoil are in good accordance with the CFD results.

Zhang et al. [24]used CNN to decode airfoil shapes from images of the airfoil into discrete, ordered vector representations and used these representations to train for the aerodynamic coefficients such as drag coefficient and lift coefficient. The first algorithm they tried provided the flow conditions as additional inputs to the airfoil representations, meaning a concatenated vector of decoded airfoil parameters and flow conditions, giving an output of lift coefficient. This algorithm did not provide promising results, so they chose another algorithm. In this algorithm, they provided the angle of attack by tilting the airfoil upwards and using that image as a separate input. The part corresponding to the airfoil's inner section in these images is entirely black. The airfoil surface is given with some transition to the freestream, which is entirely white to light gray. They used non-dimensionalization to adjust for various Mach numbers, the maximum Mach number being entirely white, and as it gets smaller, the color darkens.

The authors chose XFOIL to obtain aerodynamic coefficients of the training airfoils at various angles of attack, Reynolds numbers, and Mach numbers. They chose to implement predictions for only the lift coefficient. Negative angles of attack of the airfoils were provided by flipping the sign on both the angle of attack and the lift coefficient to extend the training samples.

They have observed that the CNN approach works significantly faster than the MLP approach with the same amount of epochs at the cost of increased computational time for a single epoch. It was shown that a CNN approach for predicting the lift coefficient is a good approach with some corrections. It was seen that this algorithm could predict unseen airfoil shapes with considerable accuracy.

4.3 Conclusion

After examining the literature, we have decided that an MLP-type neural network coupled with CST would be best suited for this study. CNN-type neural networks provide excellent results when predicting flow fields, but ANNs are more suitable for coefficient predictions since they do not lose accuracy while pixelating airfoil surfaces, especially when coupled with a suitable parametrization method. As discussed in Chapter 3, CST parametrization provides accurate airfoil representation with fewer parameters which makes it more suitable for neural network studies.

Table 4.1 Review summary of some studies in the literature

Author	**Data source**	**Flow regime**	**Shape parametrization method**	**Learning method**	**Output**
Suresh et al.	Semi-empirical data	Various pitch angles	Single airfoil	RNN	C_z
Santos et al.	XFOIL & inviscid CFD	Subsonic (XFOIL), Transonic (Inviscid)	Sobieczky's	MLP	C_d, C_l
Balla et al.	Inviscid CFD	Ma: 0.3 to 0.9	NURBS, POD for $\mathrm{C_p}$	ANN	C_p graphs
Xiaosong et al.	Viscous CFD	Ma : 0.6 to 0.7 Various Re	modified B-Splines	MLP	C_d,C_l, C_p graphs
Chen et al.	RANS	Subsonic, AoA 2°-15°	Hicks-Henne bump functions	CNN	C_d, C_l
Duru et al.	RANS	Transonic, AoA -10° to 20°	Distance map	CNN	Flow field
Zhang et al.	XFOIL	Re: 30,000 to 6,500,000 Ma: 0.3 to 0.8 AoA: -10° to 30°	Convolution	CNN	C_l
Jin et al.	Laminar CFD	Low Re	Convolution	CNN	Velocity field

CHAPTER 5

NEURAL NETWORK STUDIES FOR THE PREDICTION MODEL

We mentioned that we need a prediction methodology in Chapter 1. Various types of machine learning algorithms were used in preliminary studies, like random forests, best subset selection, and neural networks. The results from these studies and results presented in the literature indicate that neural networks are the most suitable of all the methods we researched. In this chapter, neural network studies conducted in this study are presented.

5.1 Problem definition

In this section, we will present the results of an MLP model constructed with the already existing data used in [23]. This data set was generated using an in-house CFD solver employing Reynolds-Averaged Navier Stokes equations with Spalart-Allmaras [25] turbulence model. The Mach number was fixed to 0.7, and the angle of attack ranged from -10° to 20°. 204 airfoils from the UIUC database of different shapes, airfoil families (laminar, high lift, highly cambered, supercritical, etc.), and usage areas were solved while generating the database. This was done to increase the robustness of the model while predicting new sets of airfoils. However, this also means that the solution database is highly nonlinear since it includes a post-stall region and is in a transonic regime. In the end, 6324 CFD runs were completed. CST parameters of the airfoils in the dataset are calculated using ordinary least squares, the method presented in section 3.3.1.

While separating the training and testing sets, airfoils were exclusively selected to include all kinds of airfoils; thick, thin, highly cambered, etc., and dissimilar to the training set. This selection aims to test the model’s robustness for unseen airfoil types and monitor if overfit occurs. Therefore, we can better judge the model while examining the validation set.

The data was split 10%-90% as the validation set and training set. A set of new and arbitrary airfoils were generated with the CST parametrization method and were solved in the same flow conditions to be used as the test set. The arbitrary numbers are generated using a weighted linear combination of existing airfoils and tweaking the resultant airfoils. This step ensures that a realizable airfoil shape is generated while maintaining arbitrariness. Visualizations of test set airfoils are given in the result comparison section.

The validation set was chosen manually to include airfoil shapes of all kinds (laminar, thick, thin, highly cambered, inverse design, etc.) to evaluate the model's performance for a wider portfolio. The validation set consist of following airfoils: [*e396, e544, fx75141, goe493, mh94, naca63012a, mrc20, naca1412, naca2421, naca63206, naca63412, naca64008a, naca64108, naca64209, naca642215, rae5214, raf69, rc510*].

Some data pre-processing is needed to be used as an input layer to the MLP algorithm. The input layer is prepared as a matrix where each row corresponds to a data point, and each column corresponds to the values of the predictors. The predictors, in this case, are the CST parameters and the angle of attack. The output layer is the selected aerodynamic coefficient to be trained.

Table 5.1 An except from the dataset

vU0	**vU1**	**...**	**vL4**	**vLEl**	**zLEl**	**AOA**	**CL**	**CD**	**Airfoil**
0.2099	0.1215	...	-0.0298	0.0863	-0.0009	-10	0.1049	0.0989	ah93w215
0.2099	0.1215	...	-0.0298	0.0863	-0.0009	-9	0.1072	0.0861	ah93w215

5.2 Multilayer Perceptron

The type of network used in this study is called Multi-layer Perceptron (MLP). Its structure was inspired by the firing of synapses of biological neurons. It is a feedforward neural network with at least three layers: the input layer, at least one hidden layer consisting of neurons, and an output layer. The neurons in the context of MLP use a nonlinear function called the activation function that maps the weighted inputs into the neuron to the output. The output is then multiplied by weight before being fed to the next layer. The learning is conducted with a backpropagation method that updates the weights at each epoch. A visualization is presented in Figure 5.1.

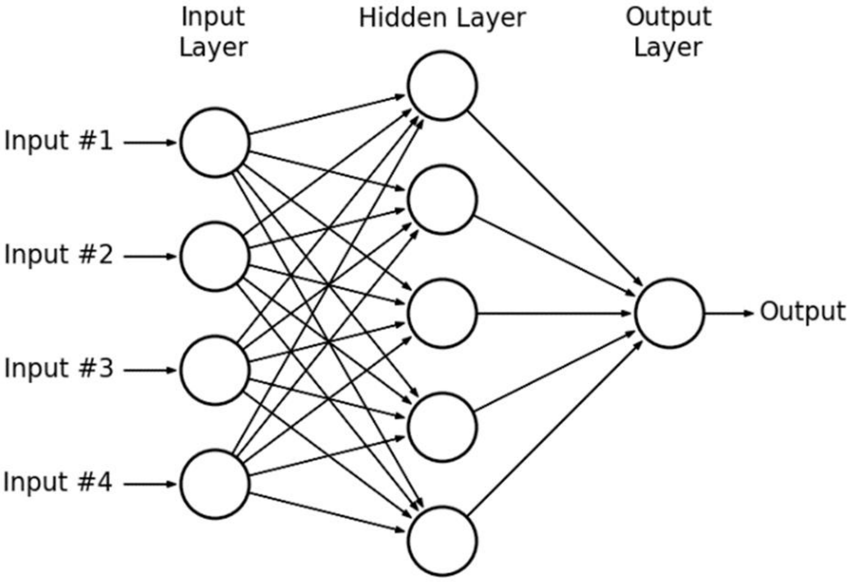

Figure 5.1 MLP visualization

Many readily available packages exist for setting up an MLP, and it is an ever-growing community. This study was conducted using the Keras package from the Tensorflow module (version 2.9.1). This package is easily adaptable in Python and easy to use.

A normalization layer is present between the input layer and the first layer of neurons. The normalization is done by centering the data around the mean and dividing by the standard deviation, as seen in Equation 5.1. This step scales the input layer as a distribution centered around 0 with a standard deviation of 1. This step is instrumental if the scales of the inputs vary significantly, as is the case with CST parameters compared to the angle of attack or Mach number. It is also a good idea to scale the output by a couple of magnitudes since some of the coefficients are small. This will help with reducing floating point errors and increase learning performance.

$$x_{new} = \frac{x - \mu}{\sigma} \qquad 5.1$$

A typical MLP model with four hidden layers constructed with Keras can be seen in the excerpt in Table 5.2:

Table 5.2 MLP construction excerpt

```
model = tf.keras.models.Sequential()
model.add(tf.keras.layers.Normalization())
model.add(tf.keras.layers.Dense(units = 1000, activation =
'relu'))
model.add(tf.keras.layers.Dense(units = 500, activation =
'relu'))
model.add(tf.keras.layers.Dense(units = 250, activation =
'relu'))
model.add(tf.keras.layers.Dense(units = 125, activation =
'relu'))
model.add(tf.keras.layers.Dense(units = 1))
model.compile(optimizer = 'adam', loss = 'mae')
history = model.fit(np.array(X_train), y_train, epochs = 500
,validation_data=(X_val,y_val))
```

The `X_train` array is the data in Table 5.1 up to, and including the AOA column, the `y_train` array is CL column. The validation dataset is given with `X_val` and `y_val` arrays in the same format. The epoch number should be selected carefully. Typically, the training should be stopped when the validation losses increase because the model starts to overfit. An example of this phenomenon can be seen in Figure 5.2. The number of epochs corresponding to that point highly depends on the number of neurons, the number of data points, the nature, and the learning rate. It is a good idea to save the model frequently, inspect when overfit occurs, and prevent it by loading a version before the overfit.

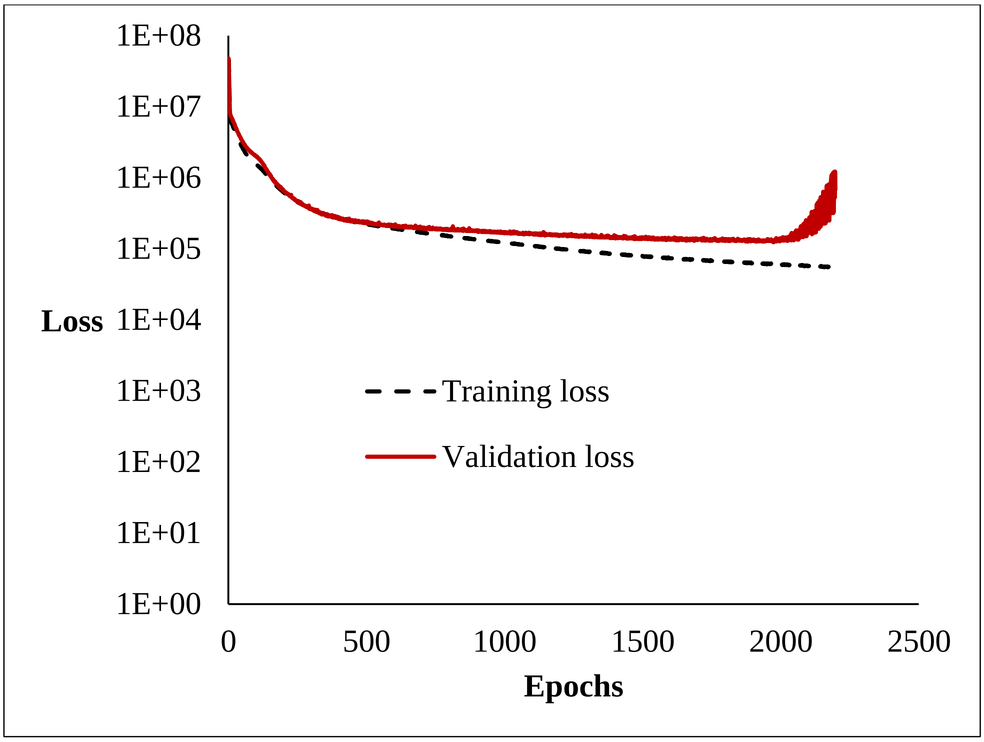

Figure 5.2 Typical overfit graph

A learning rate scheduler is given to the Adam optimizer in this study. The learning rate specifies how much of the update to the weights is used in each iteration. It is an excellent idea to decrease the learning rate near convergence since a bigger learning rate might cause zigzagging around the minimum and stall convergence. An exaggerated version of a learning rate scheduler can be seen in Figure 5.3.

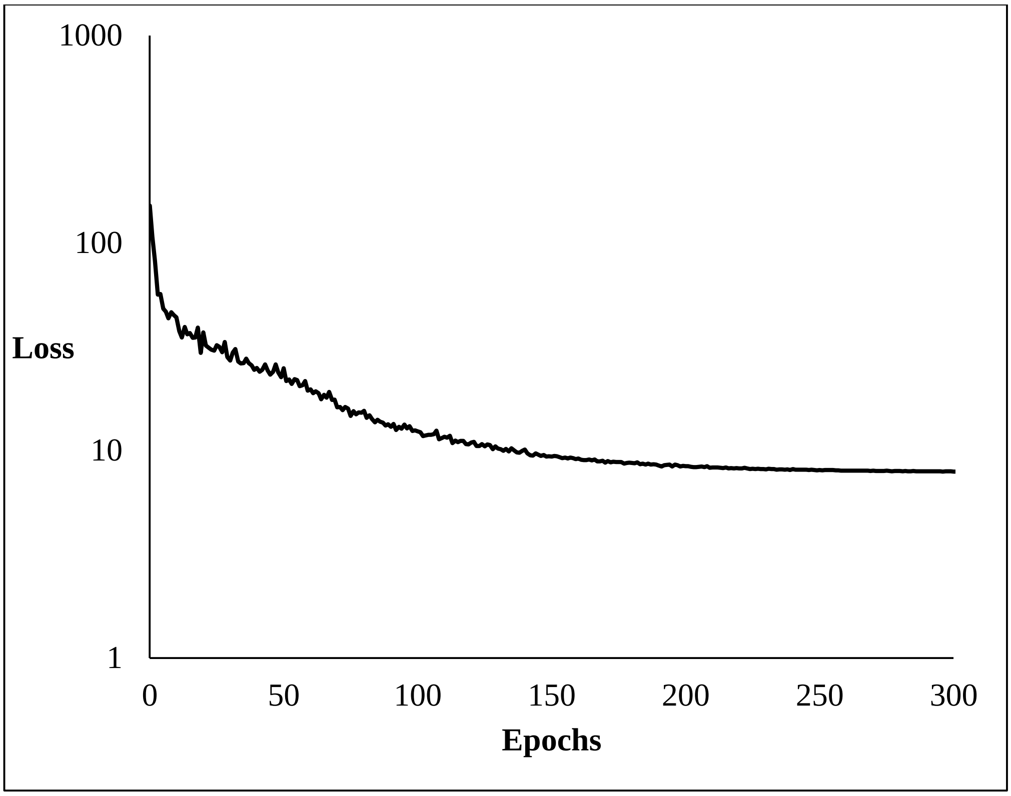

Figure 5.3 Learning rate scheduler effect on training

Table 5.3 Learning rate with Keras excerpt

```
lr_schedule                                                    =
tf.keras.optimizers.schedules.ExponentialDecay(
    initial_learning_rate=1e-1,
    decay_steps=1000000,
    decay_rate=0.99)
adam = tf.keras.optimizers.Adam(learning_rate = lr_schedule)
model.compile(optimizer = adam, loss = 'mae')
```

With the excerpt in Table 5.3, learning rate scheduler with an exponential decay can be incorporated to the learning process. The 3 parameters' effect on the learning process; initial learning rate, decay steps, and decay rate is highly dependent on the data and the user should tune them to their application.

5.3 Prediction model structure

The training model is a multi-layer perceptron created with the Keras package in Tensorflow. The model consists of 4 hidden layers with the rectified linear unit as the activation function. The number of neurons is halved after each hidden layer which can be seen in Figure 5.4.

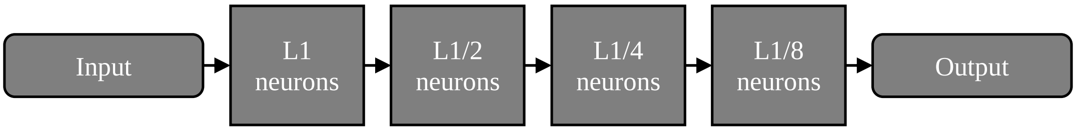

Figure 5.4 Structure of the MLP network

We employed Adam [26] as the optimizer to compute weights on the model. This optimizer provides fast convergence without sacrificing robustness. A sample of the training history can be seen on Figure 5.5.

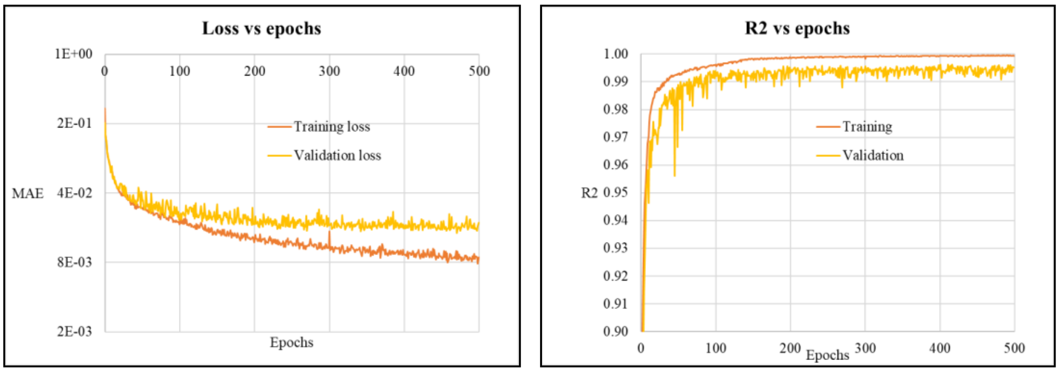

Figure 5.5 Training history of lift coefficient model

In this study, two parameters were fine-tuned to study model performance: model complexity and transformation complexity. The lift coefficient is seen as the more challenging to predict due to stall and shock location variations in the dataset. Therefore, the models studied with different parameters are presented in Table 5.4 with the lift coefficient metrics.

Table 5.4 Performance metrics for different model complexities and transformation complexities

	# of Neurons	# of CST parameters	MSE	MSE_val	R2	val_R2
Model1	1875	15	88.66	**182.16**	0.9994	**0.9951**
Model2	937	15	97.77	187.06	0.9993	0.9924
Model3	467	15	108.94	201.97	0.9991	0.9919
Model4	232	15	145.87	209.03	0.9980	0.9940
Model5	1875	17	87.39	199.32	0.9995	0.9917
Model6	1875	19	**78.38**	208.75	**0.9996**	0.9908
Model7	1875	21	85.91	216.20	0.9995	0.9842
Model8	937	21	101.55	239.96	0.9992	0.9828
Model9	467	21	106.84	199.87	0.9991	0.9904

While increasing the number of neurons in the model increases training R^2 and decreases the loss function further, it does not always translate to the test set. After evaluating different models, we can find the optimum number of CST parameters and neurons.

While examining Figure 5.6 and Figure 5.7, we can see that the model performance seems to decrease as transformation complexity increases. Intuitively, we would expect higher-order transformations to yield better results since such transformations will describe the airfoil more completely. However, this is not the case. It is possible that increasing the shape transformation complexity requires more data to perform at the same level.

Going forward with the 15-parameter model, we can study the effect of the number of neurons on the model performance.

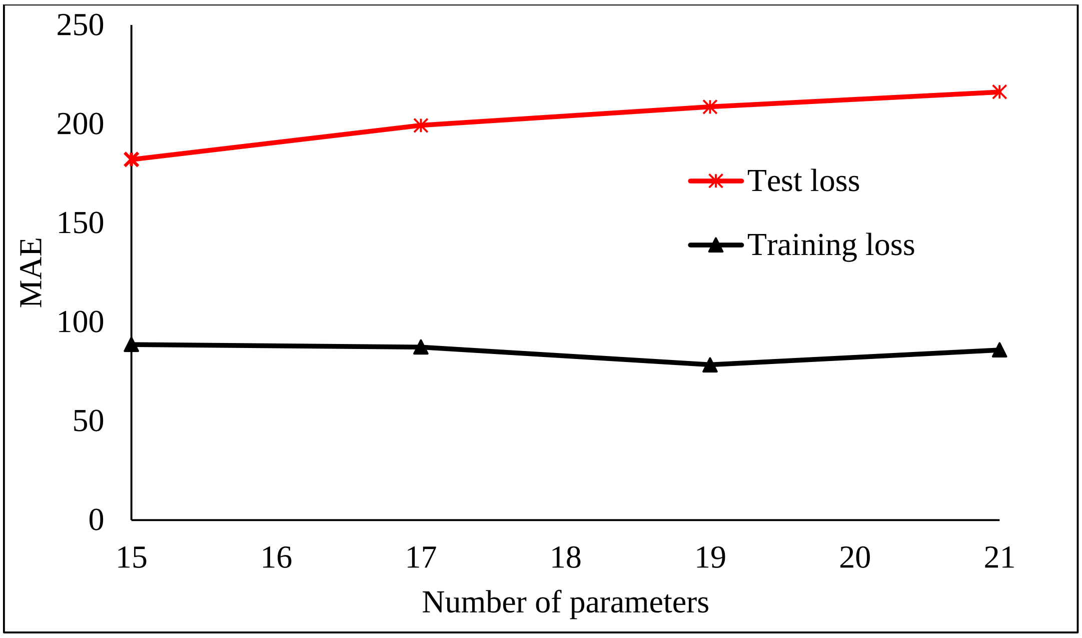

Figure 5.6 Loss vs. transformation complexity

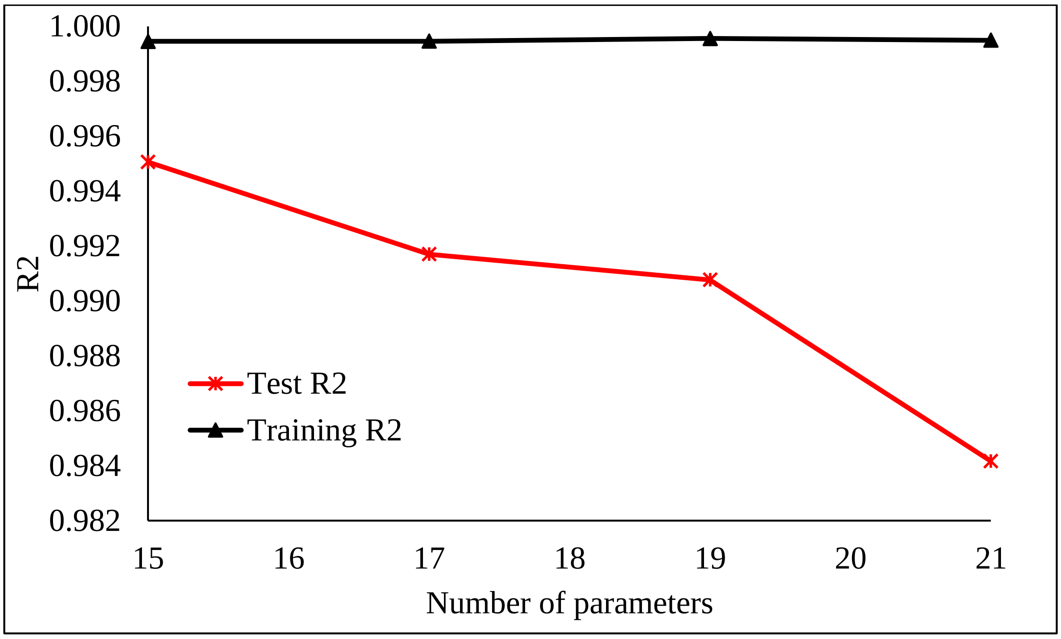

Figure 5.7 R2 score vs. transformation complexity

After examining Figure 5.8 and Figure 5.9, Model1 from Table 5.4, with 15 parameters and 1875 neurons, is chosen as the baseline model. It provided the highest R2 score and the lowest MAE and RMSE. Using more neurons than in this case shows that we achieve overfit. Remember that the number of neurons gets close to the number of data points at the overfit point, which is not a recommended practice for most applications. Examining the test airfoil data, we can see that the model provides a good correlation between CFD output and model output even though it did not train for these airfoils. Correlation plots of this model can be seen in Figure 5.10.

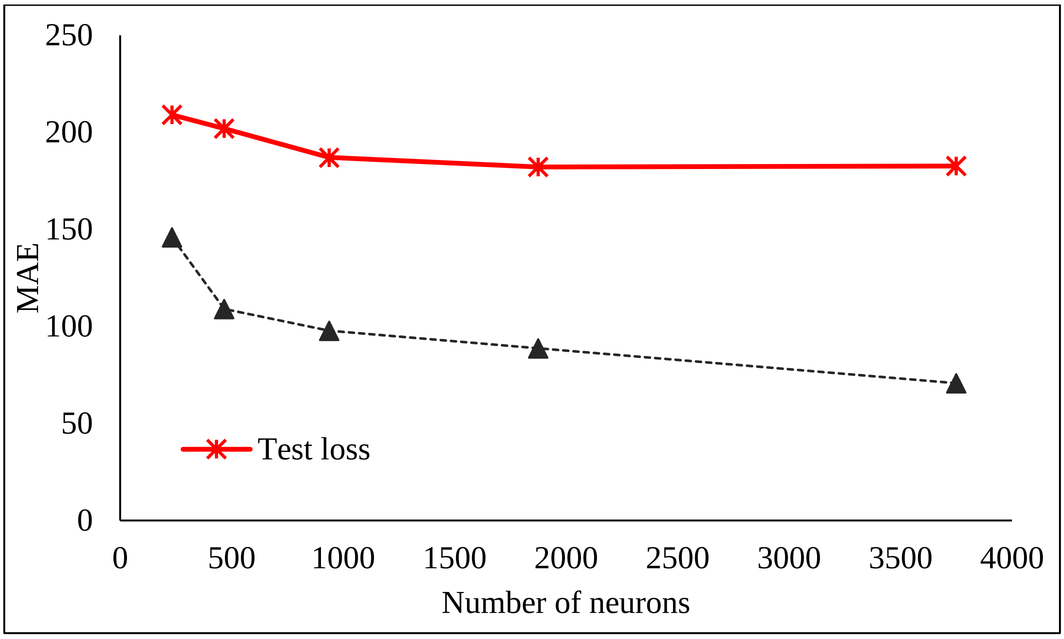

Figure 5.8 Loss vs. model complexity

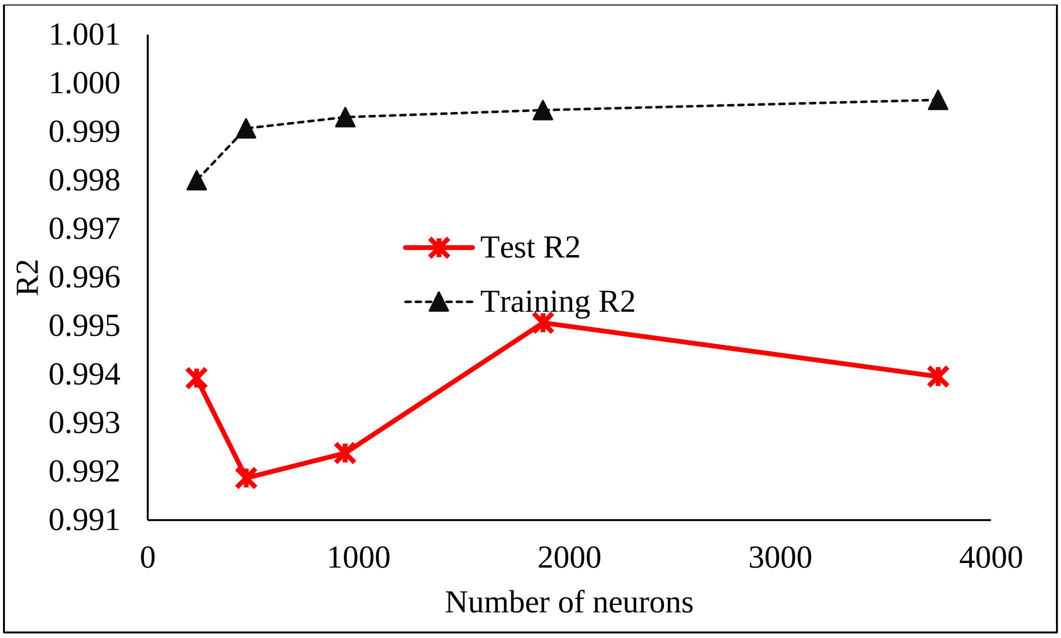

Figure 5.9 R2 score vs. model complexity

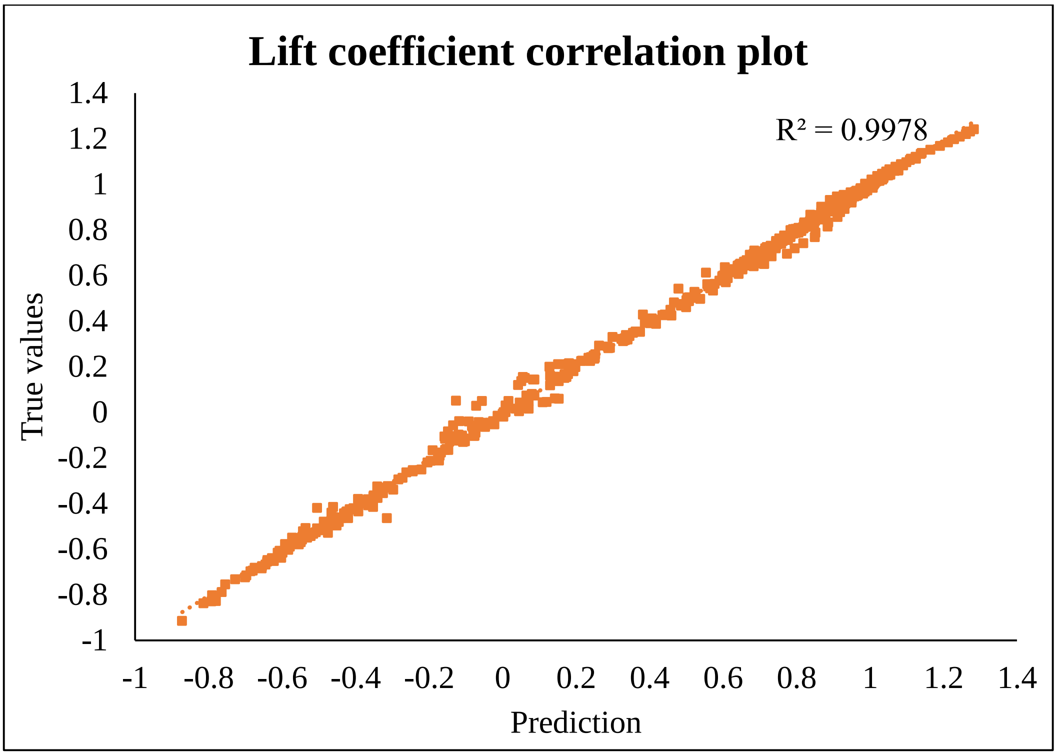

Figure 5.10 Correlation plots of the MLP model

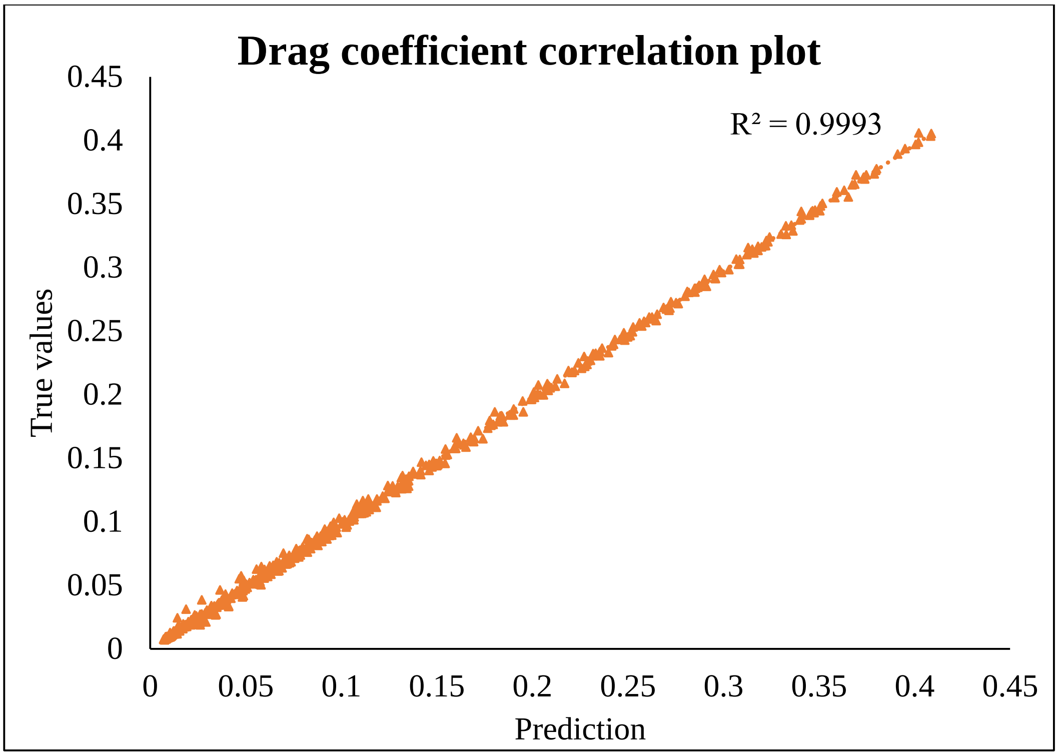

Figure 5.10 Correlation plots of the MLP model

The correlation plots show good agreement between the predictions and the CFD results. This model can be deemed successful for the validation set.

The same model structure is again tried with Bezier coefficients to compare the two shape parametrization methods. 24 Bezier coefficients and angle of attack are given to the ML model as inputs for the lift coefficient and drag coefficient outputs. The comparison is given below in Table 5.5.

Table 5.5 Parametrization method comparison for lift coefficient

	val_R2 - CL	MSE - CL
Bezier model	0.9879	435.94
CST model	0.9978	182.16

After closely examining the above plots, we can see that the CST model provides accurate results for the validation set with an R2 score of 0.9978.

This model is used to predict test set to judge its performance in a final dataset. These predictions achieved a mean absolute error of 0.047 and root mean squared error of 0.033 for C_l predictions. Given the fact that the wide angle of attack range increases the lift coefficient considerably, the test prediction errors are small. The results can be seen in Figure 5.11 and Figure 5.12. The test results are in agreement with the CFD for various airfoil shapes and families and lift changes with respect to the angle of attack are similar, with slight skewness after stall occurs. By the metrics used in this paper, we can conclude that this prediction model provides successful results.

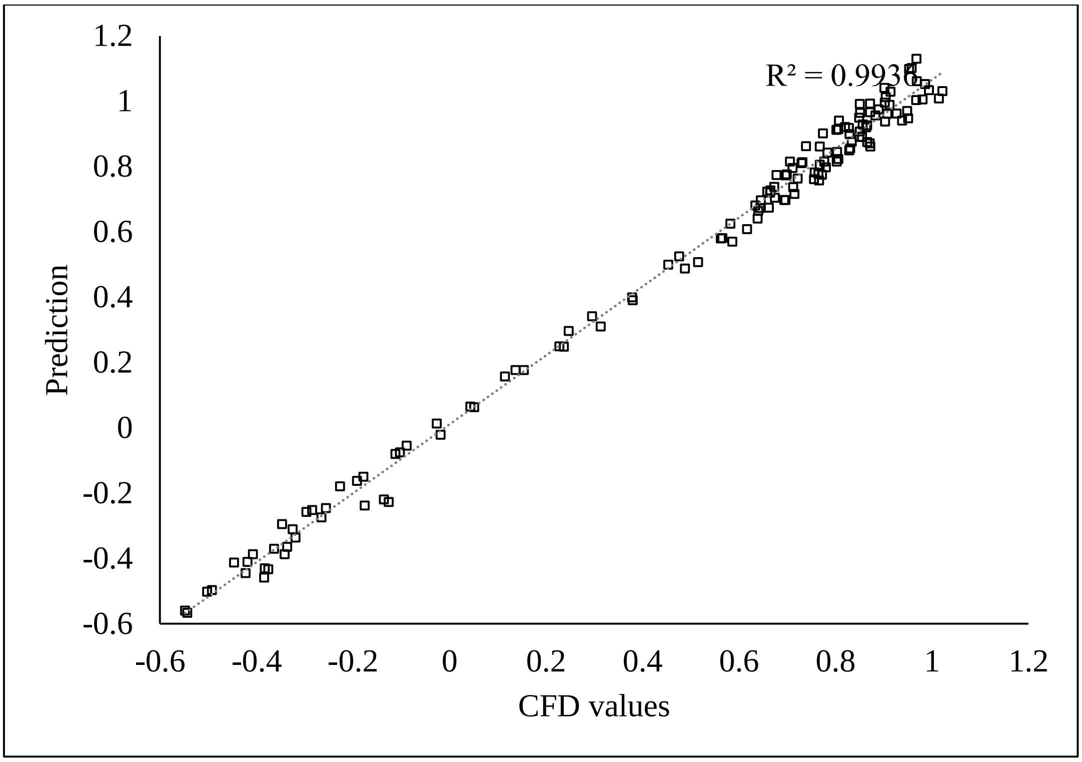

Figure 5.11 Correlation plot for test set lift coefficient

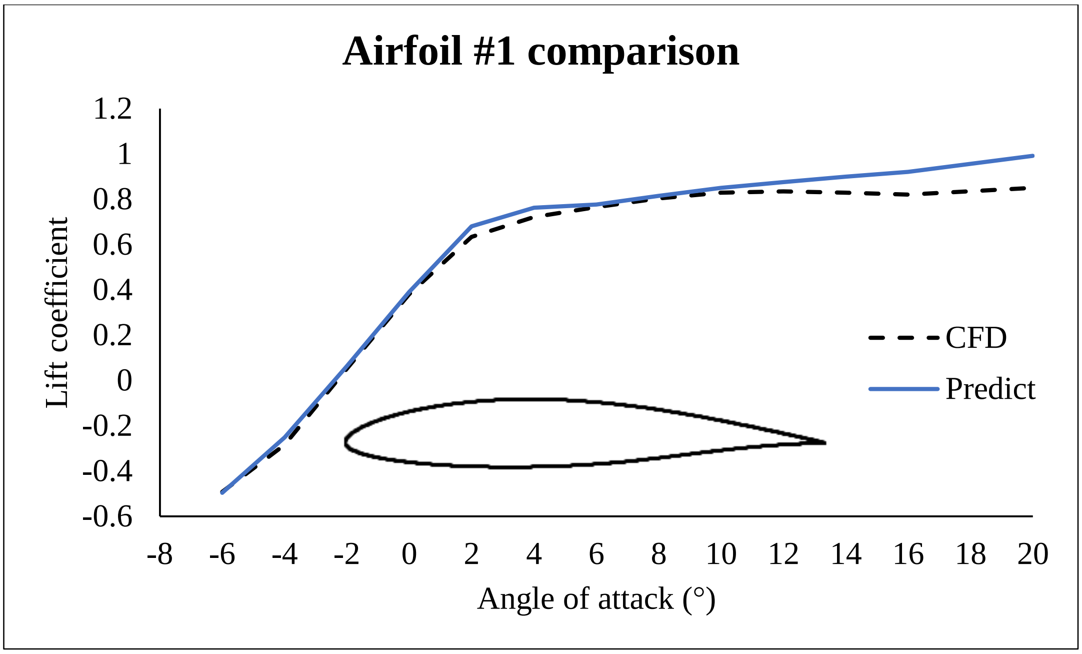

Figure 5.12 Lift coefficient comparison between prediction and CFD for randomly selected test airfoils

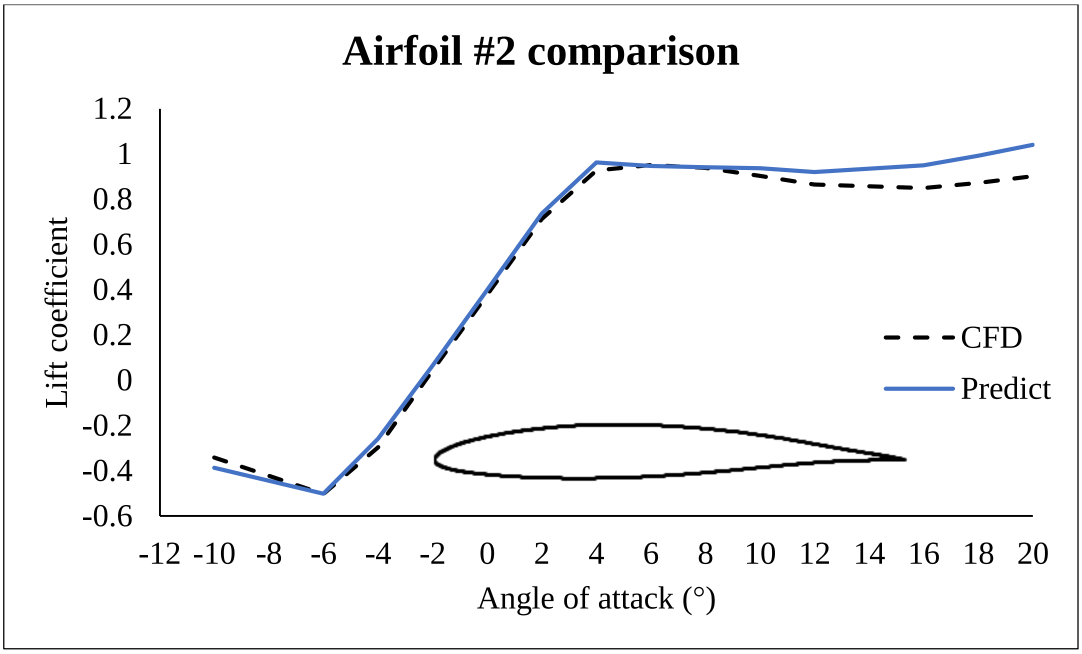

Figure 5.12 Lift coefficient comparison between prediction and CFD for randomly selected test airfoils

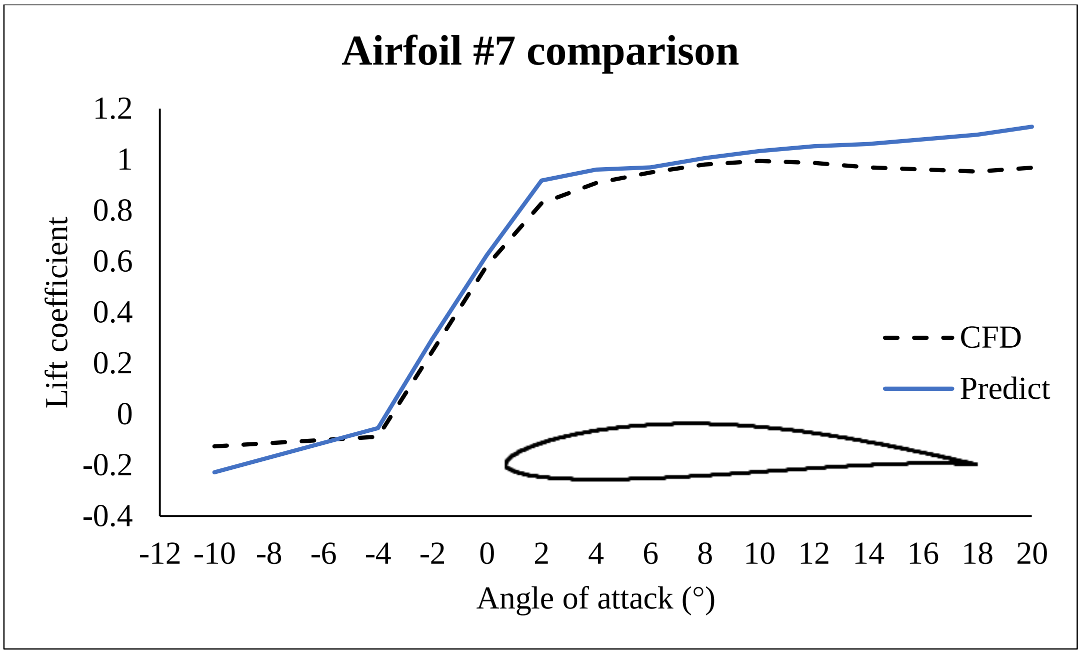

Figure 5.12 Lift coefficient comparison between prediction and CFD for randomly selected test airfoils

CHAPTER 6

DATA GENERATION

This chapter presents data generation methodology for neural network training for predicting transonic regime and validation studies regarding it. Flowpsi is used as the RANS solver with HLLC [27] flux splitting scheme. This scheme is used to capture shocks accurately while not sacrificing the solution's robustness. The k-ω-SST [28] turbulence model is employed in the solutions with a wall first layer height small enough that wall y^+ values lie between 0.1 and 1.

6.1 Mesh independence

In this study, and for the rest of the document, two-dimensional structured grids were employed. For the mesh independence study, we used NACA4412 validation study presented on the NASA turbulence model validation website [29]. This study was selected since it has a high angle of attack and has various results for different turbulence models. Flow conditions are given below.

Table 6.1 Flow conditions for mesh independence

Mach number	0.09
Reynolds number	1,520,000
Angle of attack	13.87°

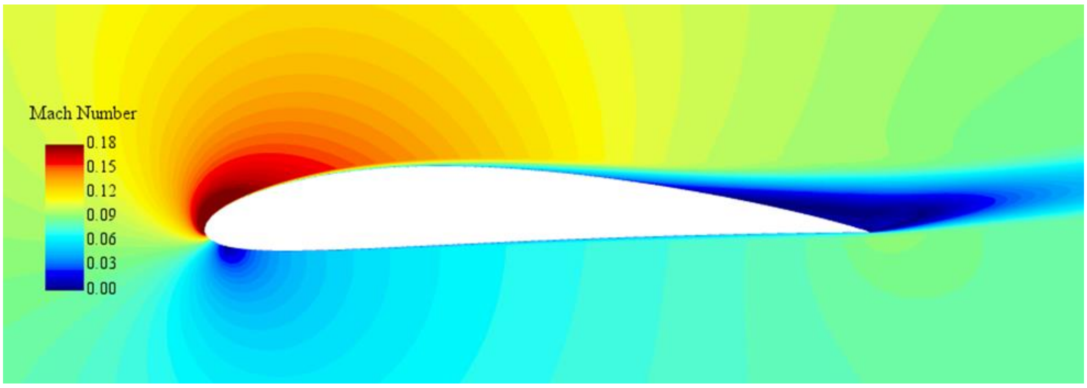

Figure 6.1 Mach number contour of mesh independence solutions

The results of the airfoil surface grid independence study can be seen in Figure 6.2. The study was conducted by increasing the surface resolution of the airfoil while keeping the farfield radius and first layer height constant. Drag count refers to the drag coefficient multiplied by 10,000. With the result of this study, we selected the surface resolution of 2012 points per airfoil.

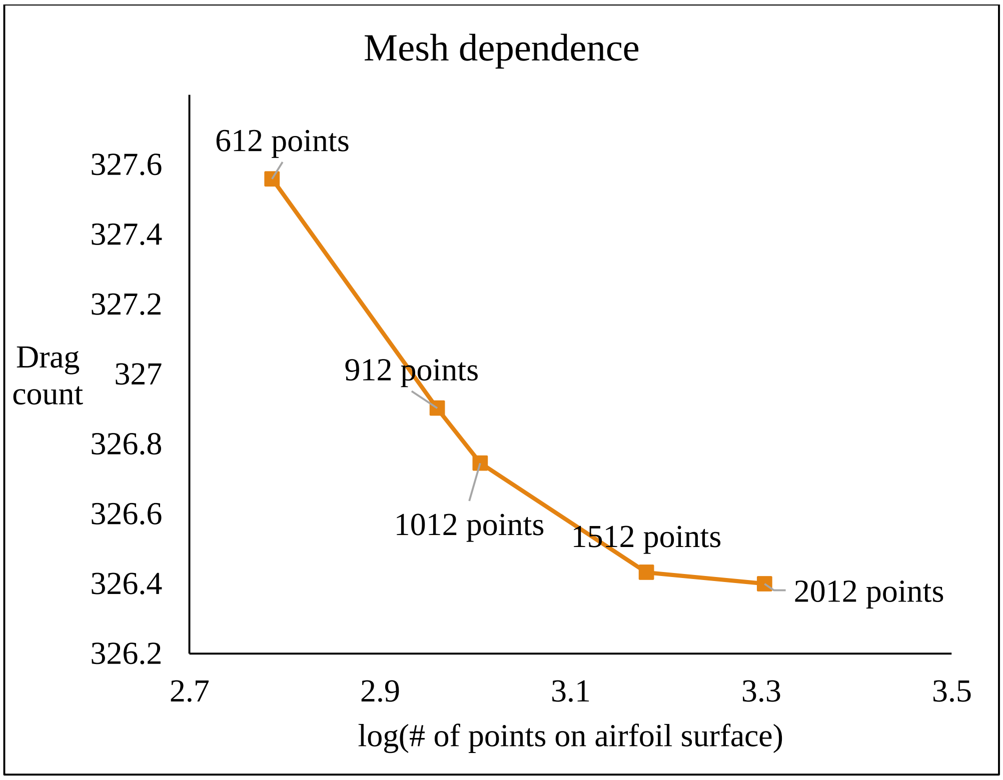

Figure 6.2 Grid dependence results.

Another study is used to decide on the farfield radius. Keeping the first layer heights, surface resolution, and mesh growth rate constant and varying the farfield radius, we can examine the effects of the farfield radius.

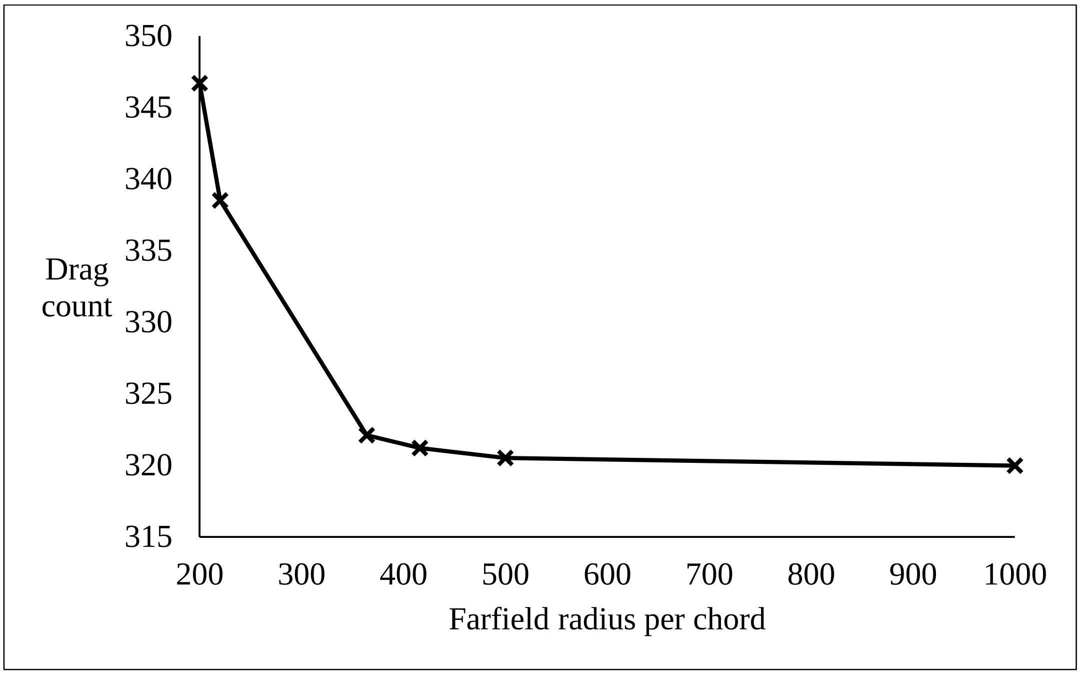

Figure 6.3 Farfield radius study results

After examining the results, with the surface resolution of 2012 points, presented in Figure 6.3, we decided on the farfield radius of 1000 per airfoil chord. The final mesh resolution is 2012x103. A close-up of the volume mesh is presented in Figure 6.6. A comparison of the NASA website results and Flowpsi can be seen in Figure 6.4.

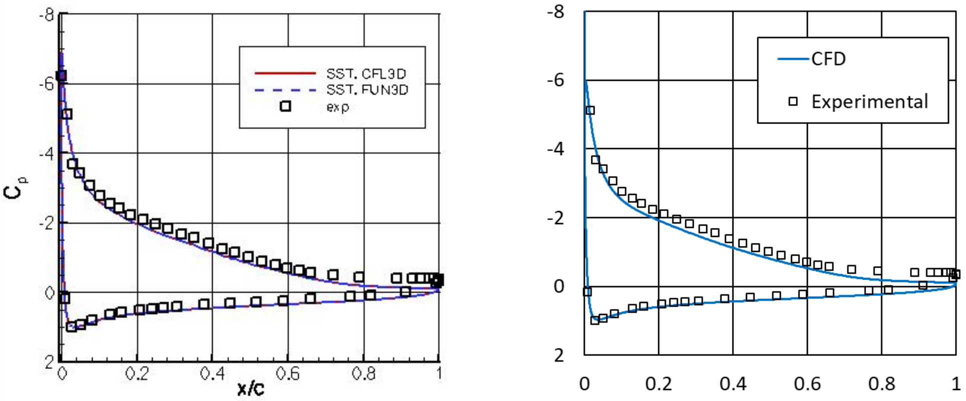

Figure 6.4 Comparison of the NASA website (left) and Flowpsi (right)

6.2 Validation study

For validation of the solver and its settings, RAE2822 airfoil solution comparison with the corrected experimental data is used [30]. It is a well-known benchmark case for transonic turbulent flow over an airfoil. For this case, the experiment flow conditions are given in Table 6.2.

Table 6.2 Validation experiment conditions

Mach number	0.729
Reynolds number	6,500,000
Angle of attack	2.92°

Comparison of two CFD solutions obtained with Flowpsi and StarCCM+ and experimental data is given in Figure 6.5. The pressure coefficient distributions are in agreement with the CFD and the experimental data. The solution methods and the mesh parameters used in this comparison is accurate enough for generating airfoil solutions in transonic flow regime with good shock accuracy. This ensures that the data used in the neural network studies is robust, accurate and consistent.

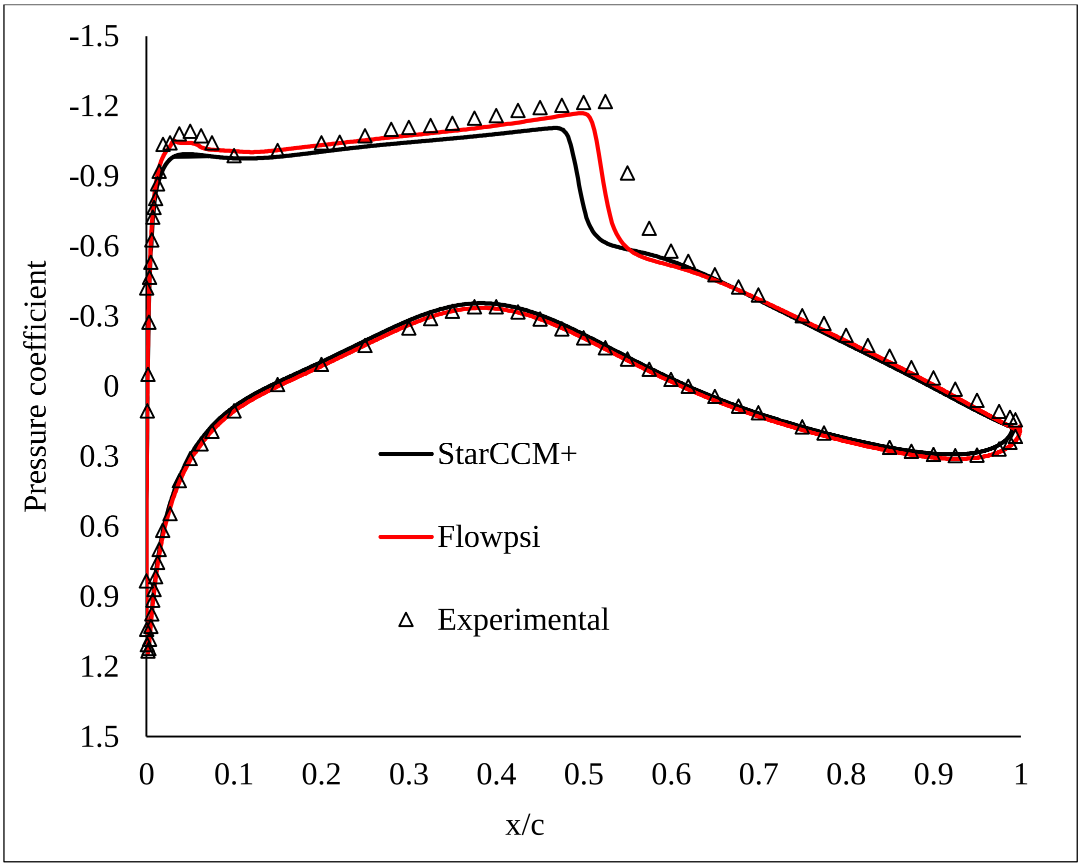

Figure 6.5. Pressure coefficient graphs of two solvers and experiment results

In the end, the final parameters used in solutions are presented in Table 6.3.

Table 6.3 Solution parameters used in this study

Parameter	**Value**
Chord length	1
Farfield radius	1000
Mesh type	Structured
Turbulence model	SST k-w
Grid size	2012x103

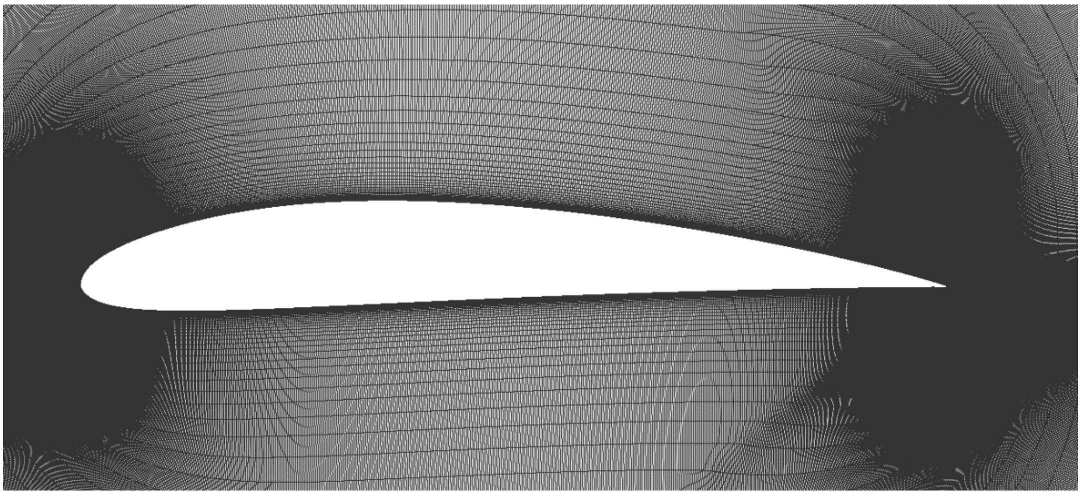

Figure 6.6 Close up view of the final mesh

CHAPTER 7

NEURAL NETWORK MODEL FOR COEFFICIENT PREDICTIONS IN TRANSONIC REGIME

In this chapter, the studies regarding the construction of the model that can predict airfoil aerodynamic coefficients are presented. This model was constructed to predict lift and drag coefficients in a variable Mach number regime. After the success of the first model in the fixed Mach number regime, the same structure is tested with varying Mach numbers as an additional input. The variable Mach number enables us to calculate $\mathrm{Ma}_{\mathrm{DD}}$ which can be used as the operation point of a supercritical airfoil.

7.1 Dataset definition

The dataset was obtained on 49 legacy supercritical airfoils and some arbitrarily generated ones with the CST method. 15% of the legacy airfoils were separated as the validation set. The remaining 85% of the legacy airfoils are the training set, and finally, arbitrarily generated airfoils were used as the test set. The angle of attack is distributed between -2° and 8°, which is a typical angle of attack range for an airliner. The Mach number is selected between 0.64 and 0.84, which most drag divergence Mach numbers lie between for supercritical airfoils. We chose the Reynolds number as 6,000,000 for all CFD runs. 9473 CFD runs in total were completed.

The CFD analyses were conducted with 14000 iterations. This is a big number of iterations since most cases converge much earlier. This was done to ensure the convergence of each case and keep the number of iterations constant for each case. After the analyses were completed, a convergence check on the aerodynamic coefficients was completed. The results were discarded if the drag coefficient or lift coefficient oscillated more than 5%.

Table 7.1 Flow conditions for supercritical CFD runs

Angle of attack	[-2°, -1°, 0°, 1°, 2°, 3°, 4°, 5°, 6°, 7°, 8°]
Mach number	[0.64, 0.68, 0.7, 0.72, 0.74, 0.76, 0.77, 0.78, 0.8, 0.82, 0.84]
Legacy airfoils	[bac1, cast10-2, cast7, cessna_mod, dfvlr, naca63a210, naca64a010, naca64a210, naca64a410, naca651213, nlr-7301, npl9510, rae100, rae101, rae102, rae103, rae104, rae2822, rae5212, rae5213, rae5214, rae5215, sc1012r8, sc1094r8, sc1095, sc1095r8, sc20010, sc20012, sc20402, sc20403, sc20404, sc20406, sc20410, sc20412, sc20414, sc20503, sc20518, sc20606, sc20610, sc20612, sc20614, sc20706, sc20710, sc20712, sc20714, sc21006, sc21010, sc2110, sc3-0712, ta11]

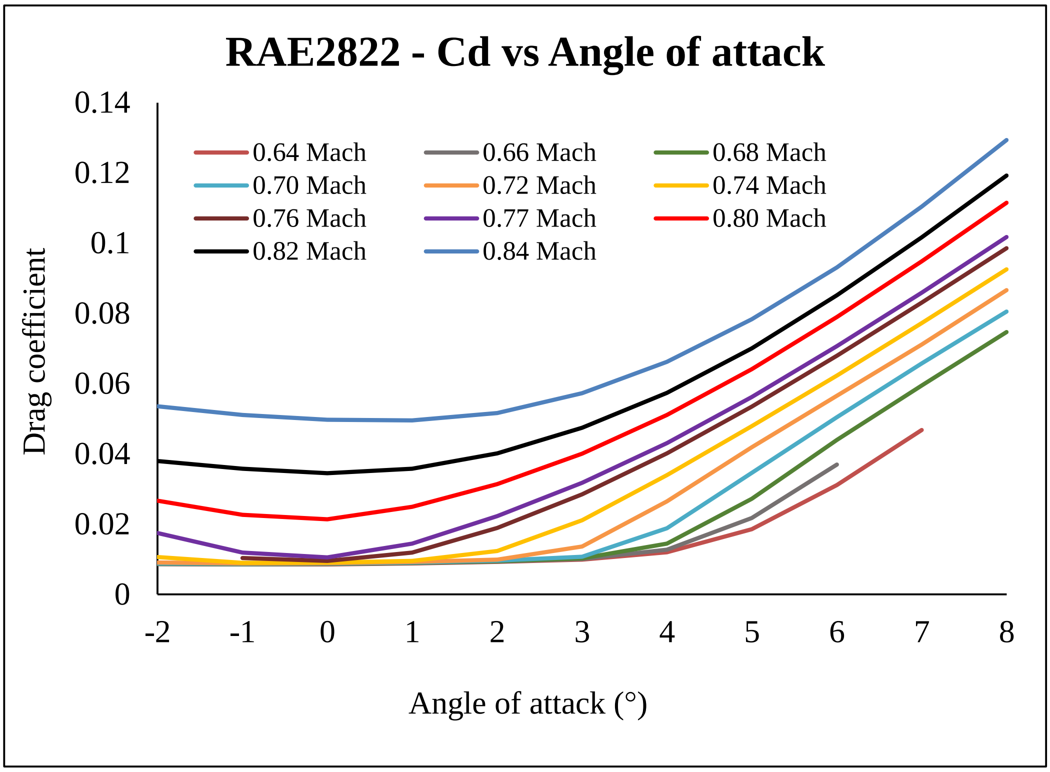

Figure 7.1 RAE2822 CFD solutions

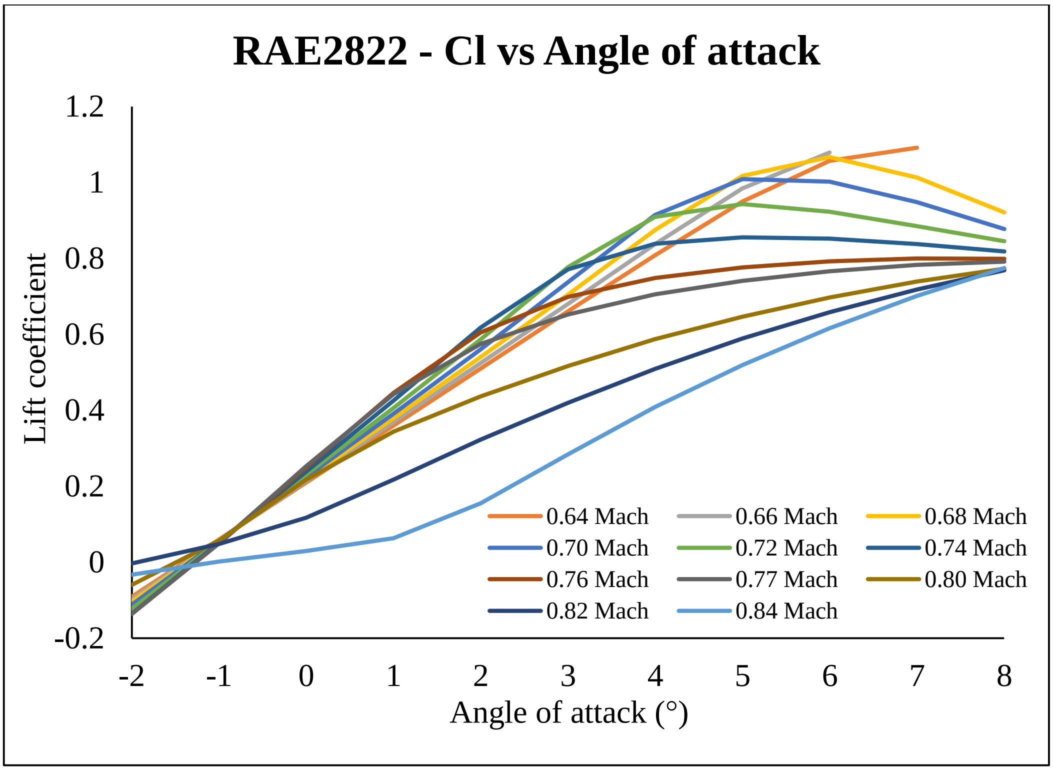

Figure 7.1 RAE2822 CFD solutions

As we can see in Figure 7.1, the data is highly nonlinear and difficult to predict with conventional reduced-order models. Every airfoil behaves differently in the same conditions since they all have different shapes. The airfoil shape dictates all these nonlinearities, and using CST is especially useful in this regime since details of the shape are inherent in the transformation. A contour plot of one of the nonlinear points is given in Figure 7.2.

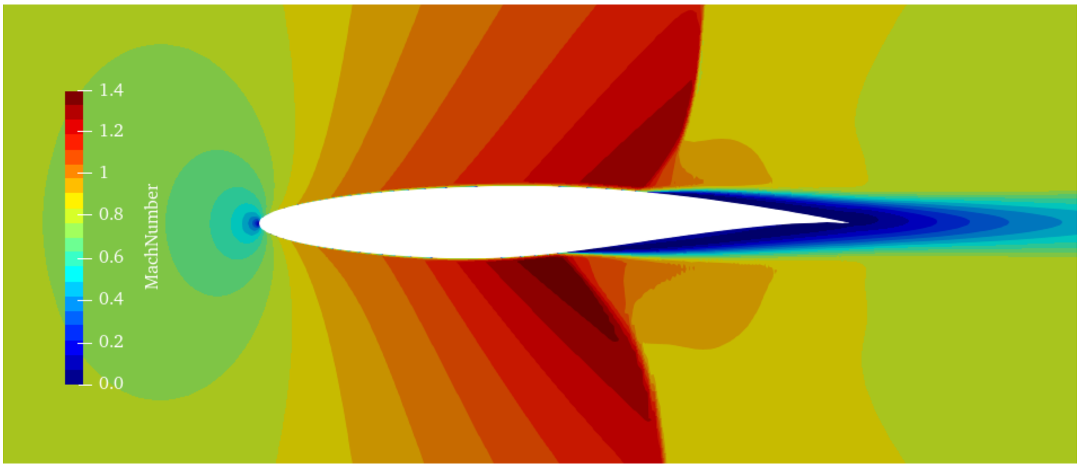

Figure 7.2 Mach number contour of RAE2822, 0.84 Mach and 0° angle of attack freestream

7.2 Model construction and results

After the data is collected and ordered in a way that can be used in neural networks, the model was constructed.

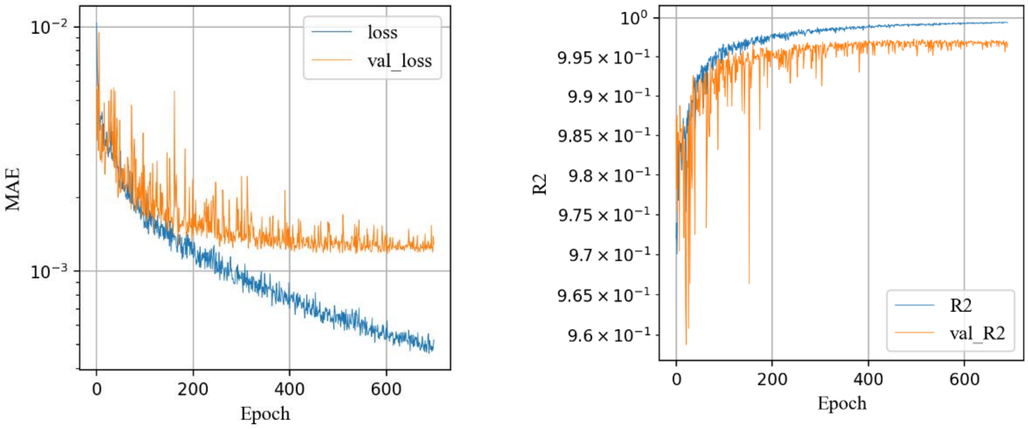

Figure 7.3 Drag coefficient training graphs

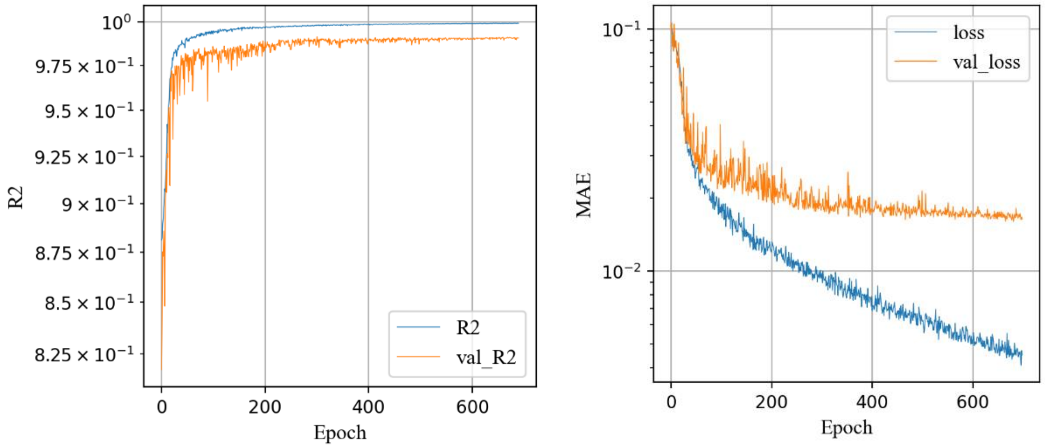

Figure 7.4 Lift coefficient training graphs

After tuning the hyperparameters of the neural network model, R2 scores of 0.9975 for the drag coefficient and 0.9935 for the lift coefficient were obtained in the validation set. The rest of the metrics can be seen in Table 7.2, and training histories can be seen in Figure 7.3 and Figure 7.4. The effects of the learning rate scheduler that is used in this study can be easily seen in these figures. The learning rate is decreased near the end, and the training is stopped before overfit occurs. Correlation plots of the validation set can also be seen in Figure 7.5**Error! Reference source not found.**. The results are in good agreement except for some region in the lift coefficient. However, these fluctuations are not seen in the test set, whose results will be presented later in this section, in Figure 7.8.

Table 7.2 Validation set metrics

Metric\Coefficient	**CD**	**CL**
R2	0.9975	0.9935
Mean absolute error	12.13E-4	16.40E-3
Root mean squared error	23.06E-4	27.27E-3

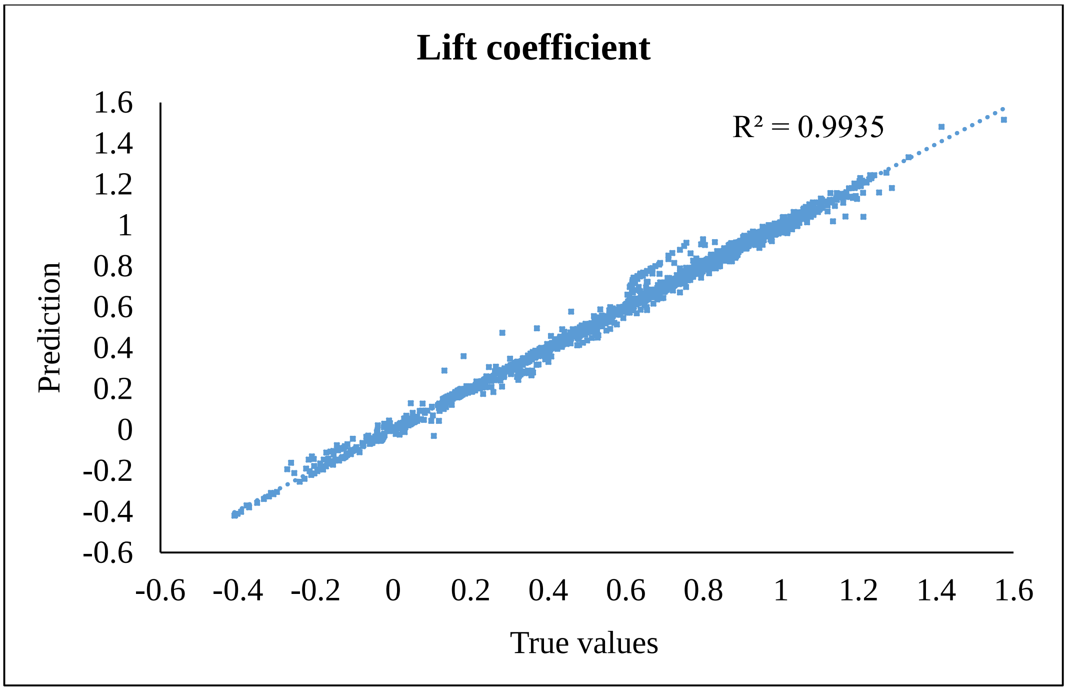

Figure 7.5 Correlation plots of the validation set

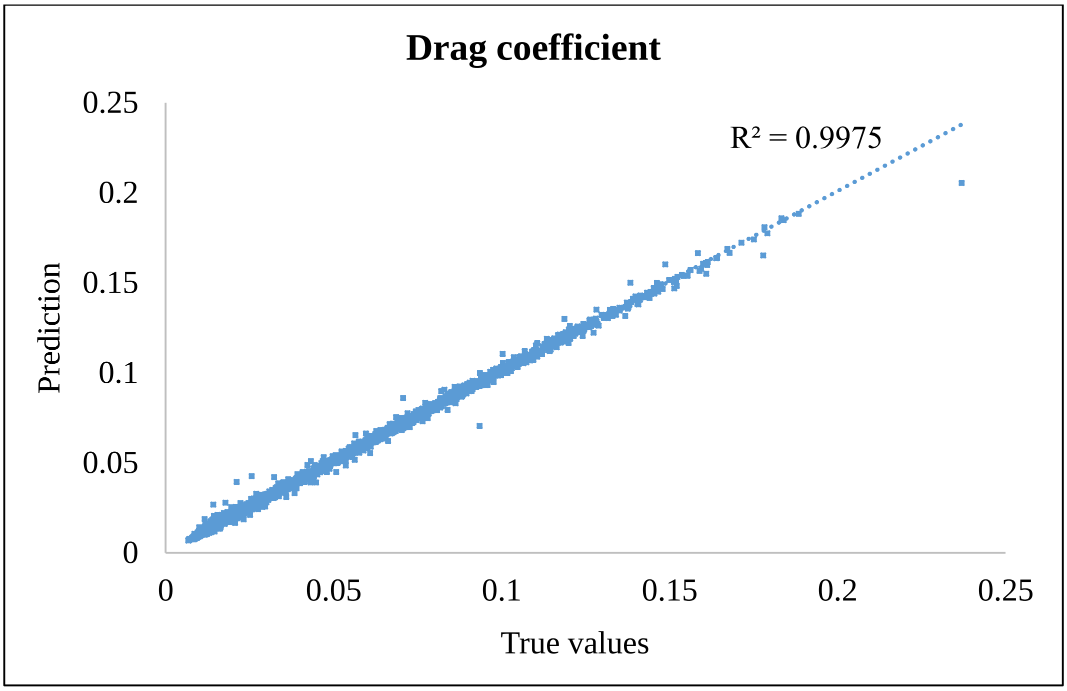

Figure 7.5 Correlation plots of the validation set

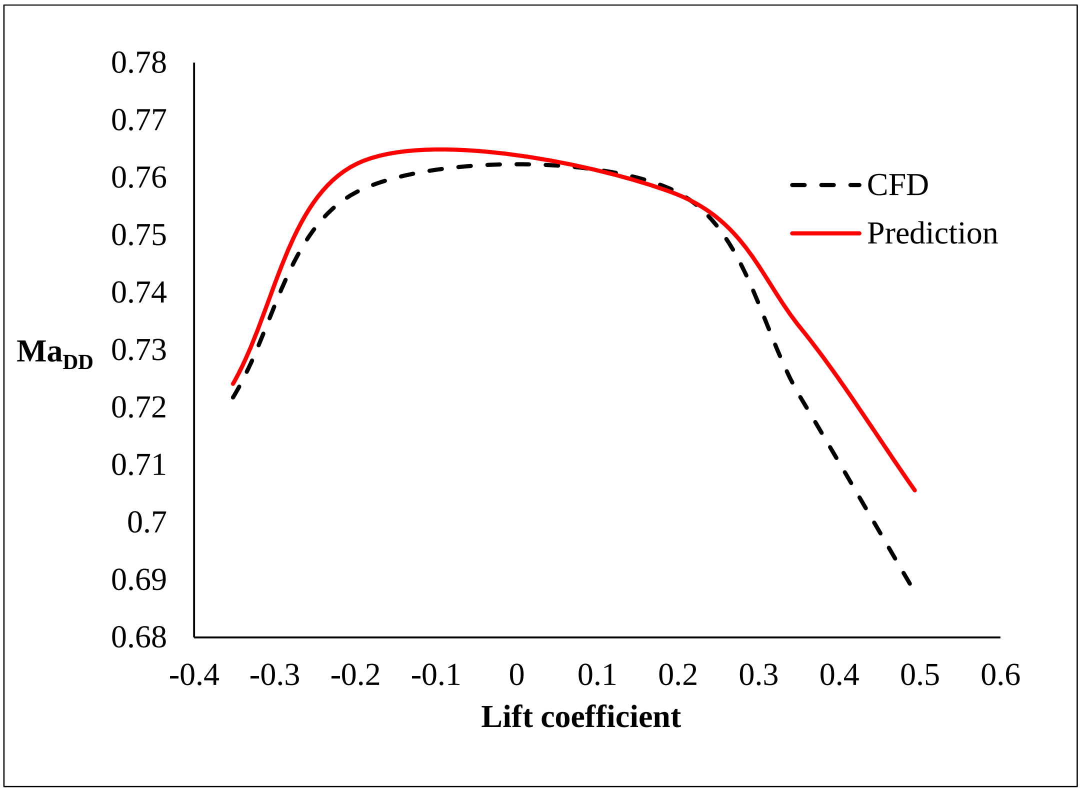

Figure 7.6 Ma_{DD} comparison for a validation airfoil

In the end, the model predictions capture the drag divergence Mach number sufficiently, even though it is a derived number from predictions. The drag divergence Mach number can be used as an output when constructing the model, but the risk of inconsistency arises between Ma_{DD} and drag coefficient predictions since they would be independent of each other. Comparisons for an airfoil from the validation set can be seen on Figure 7.7. The predictions and the CFD results are in good agreement.

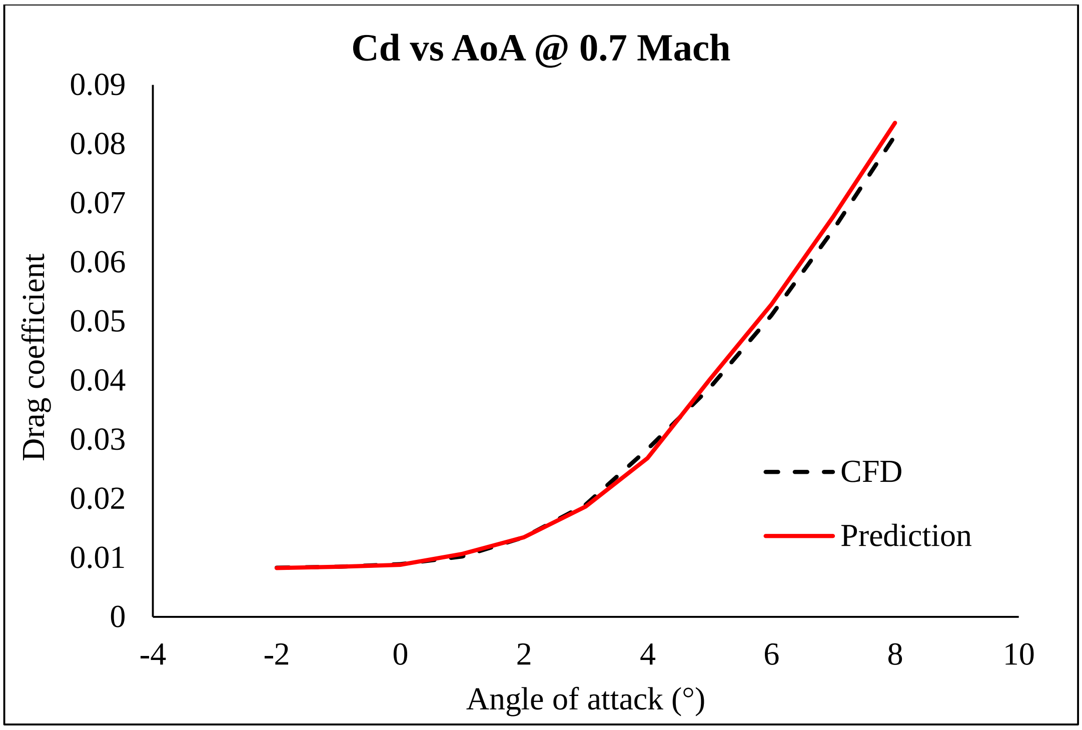

Figure 7.7 Predictions for a validation set airfoil

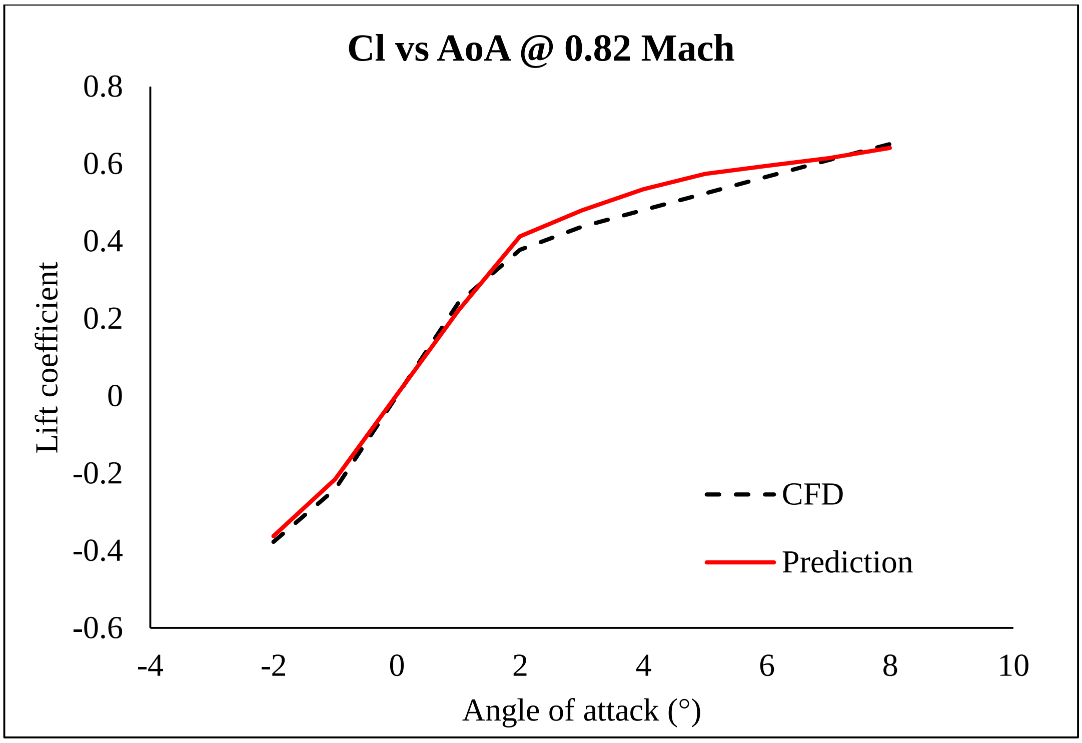

Figure 7.7 Predictions for a validation set airfoil

The model's performance is needed to be tested with the test set to evaluate its performance truly. Test airfoils were created arbitrarily within the design space defined by the legacy airfoils. These arbitrary airfoils were analyzed using CFD with the same parameters, same Mach number, and angle of attack range as the training airfoils. The results were compared with the predictions generated from the neural network model.

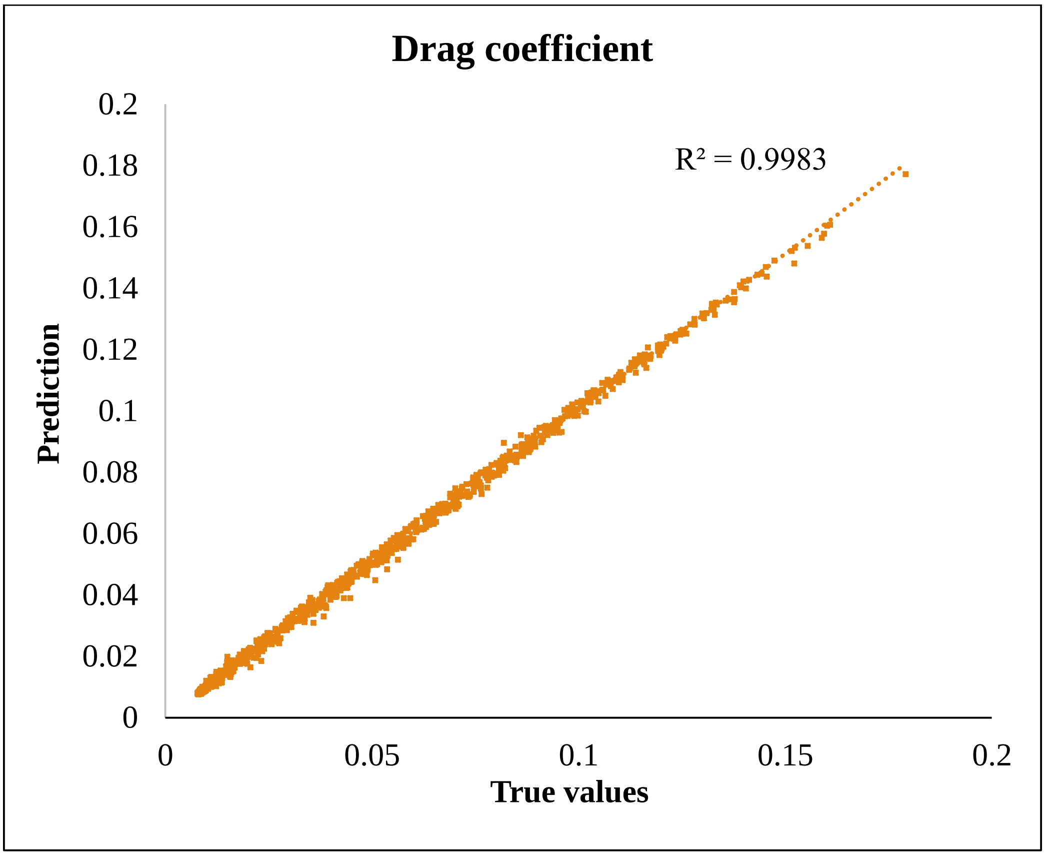

Figure 7.8 Test set correlation plots

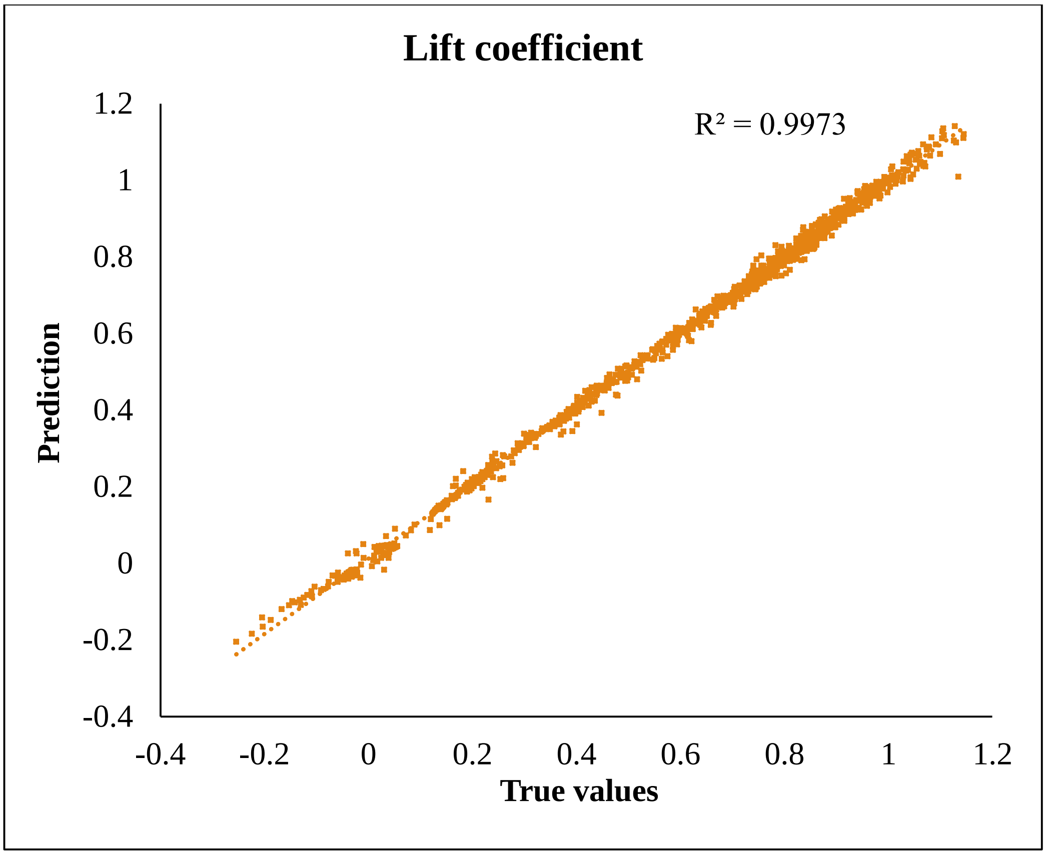

Figure 7.8 Test set correlation plots

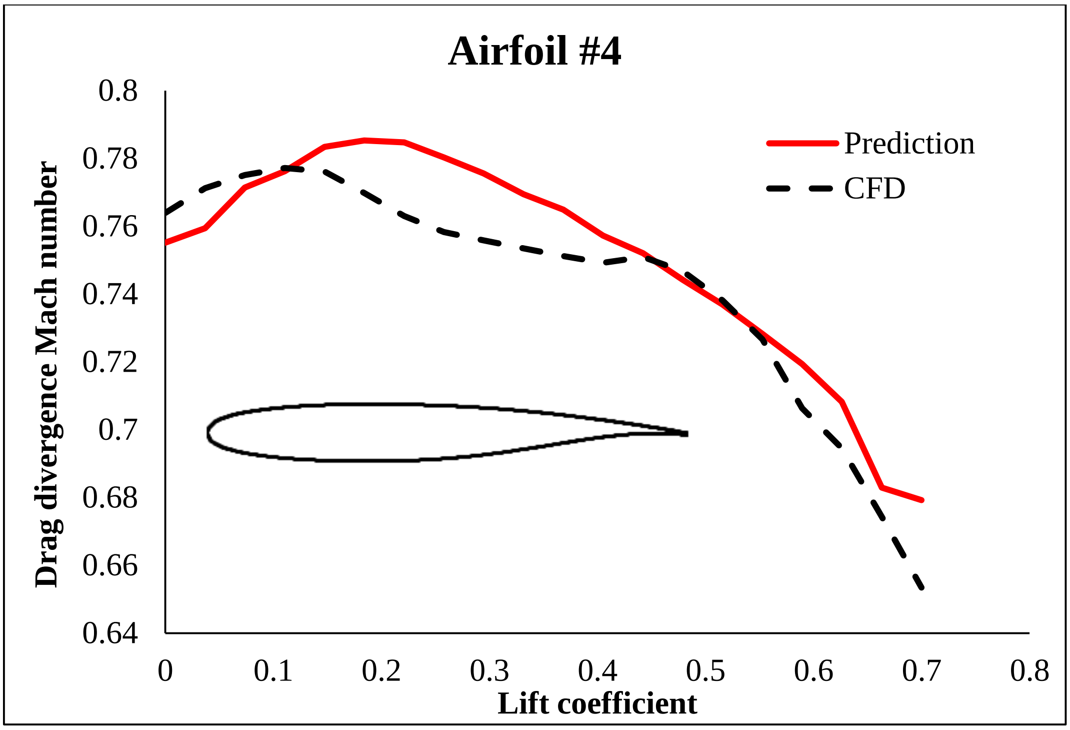

Figure 7.9 Test set results for Ma_{DD}

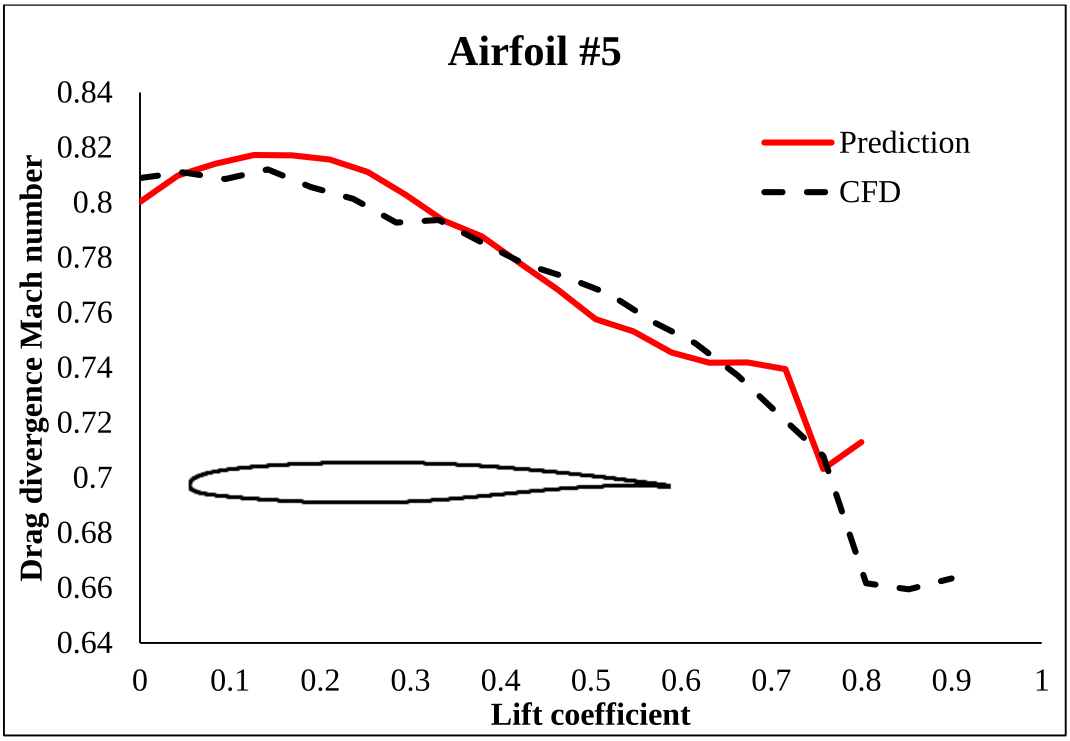

Figure 7.9 Test set results for Ma_{DD}

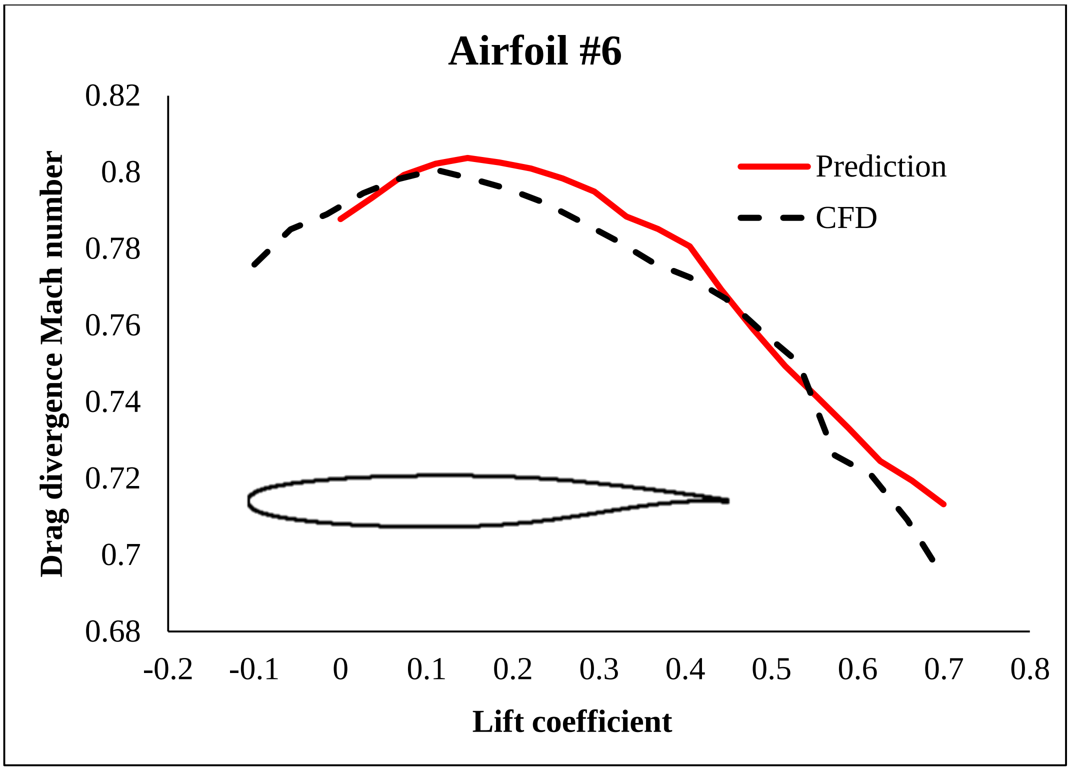

Figure 7.9 Test set results for Ma_{DD}

The test set results for Ma_{DD} are in agreement with the CFD results. The average error between the prediction and true set is around 0.01-0.02. We should keep in mind that the CFD results contain numerical errors and are not the absolute truth. The ranges in Ma_{DD} graphs change for every airfoil as their aerodynamic performance is different from each other. However, the trends between the lift coefficient and Ma_{DD} seem similar, with some shifts in both axes. CFD results and prediction results are also in agreement with the trend of these graphs.

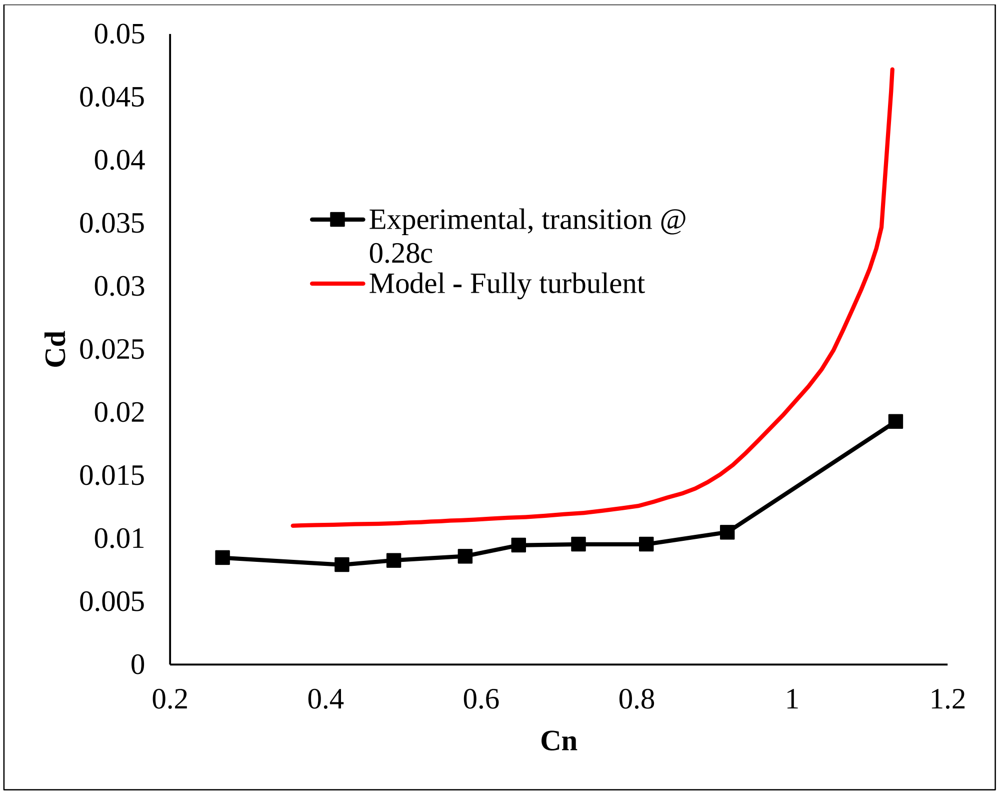

Figure 7.10 Comparison plot with experimental results [31]

A prediction comparison of the SC2-0714 airfoil with experimental results can be seen in Figure 7.10. This experiment is conducted at 6,000,000 Reynolds number and 0.7 Mach number but transition from laminar to turbulent is fixed at 28% of the chord while the CFD results are obtained fully turbulent. The results are similar to a shift in the prediction model. The reasons for this can be the 28% laminar flow, 3D effects of the wind tunnel, and wall effects of the experiment results.

Another comparison can be seen in Figure 7.11. The results are of the airfoil CAST-7, which is another supercritical airfoil and also present in the training set. The experiment was conducted conducted at 6,000,000 Reynolds number, 0.76 Mach number with transition at 6% of the chord. The results are in good agreement with slight differences. These differences can be attributed to various things, like transition at 6% or 3D effects present in the wind tunnel but overall, we can say that the results are good.

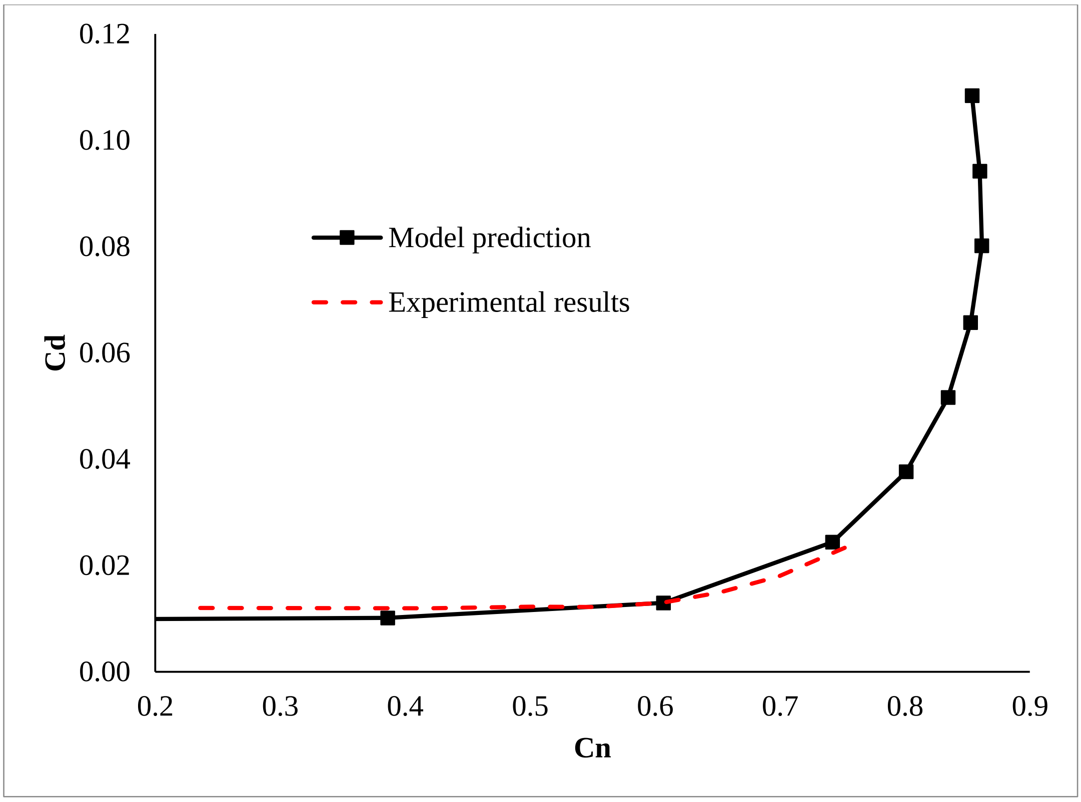

Figure 7.11 Comparison plot with experimental results [32]

CHAPTER 8

CONCLUSION AND FUTURE WORKS

In this study, we used CST to for parametric description of airfoils, that are often provided as a list of discrete points. CST proved an excellent shape parametrization method for extracting critical information from the airfoils with few parameters. These parameters and flow conditions were supplied to an MLP algorithm as the input layer, and drag and lift coefficients were selected as the output. After the parameters of the MLP algorithm were trained and hyperparameters were tuned, the model we obtained provided excellent results in predicting aerodynamic force coefficients and drag divergence Mach number, $\mathrm{Ma}_{\mathrm{DD}}$. Our results indicate that utilization of CST method allow us to obtain higher accuracy compared to other studies.

Only 2.62 seconds elapse with a 4.0 GHz processor to obtain 27000 drag and lift coefficients solutions in a range of -2° to 8 ° with the neural network trained in this study. To compare, this results in a 47x speed-up compared to XFOIL, although a speed-up over XFOIL is not the scope of this study. Also, note that accuracy of our solutions are higher than existing methods and can provide predictions in fully turbulent and transonic regime.

This study can be extended to a wider Mach number range and a more comprehensive range of airfoil shapes. The scope could be widened to include all airfoil families within a wider Mach number and angle of attack range.

To conclude, the neural network model presented in this study can be used as a low-fidelity tool that can generate results for supercritical airfoils, especially near drag divergence Mach number. The tool we obtained is faster with better accuracy, better range whereas other tools present in the literature lack performance.

For future works, a neural network model that can also predict the pitching moment coefficient, C_m, is planned. Also, a model that can predict all airfoil families and a wider Mach number range, from 0 to 1.5, is planned. Reynolds number correction for the coefficients with semi-empirical relations can be added to this study to expand the Reynolds number regime is also planned.

REFERENCES

[1] I. Abbott and A. Von Doenhoff, Theory of wing sections, New York: Dover Publications, 1959.

[2] J. Roskam, Airplane Design Part I : Preliminary Sizing of Airplanes, Design, Analysis and Research Corporation, 1985.

[3] M. S. Selig, Summary of Low-speed Airfoil Data, 1995.

[4] "UIUC Airfoil Data Site," UIUC Applied Aerodynamics Group, [Online]. Available: https://m-selig.ae.illinois.edu/ads.html. [Accessed 2022].

[5] G. E. Nitzberg and S. Crandall, "A study of flow changes associated with airfoil section drag rise at supercritical speeds," *NACA-TN-1813,* 1949.

[6] D. P. Raymer, Aircraft Design: A Conceptual Approach, AIAA Education Series, 1992.

[7] M. R. Birajdar and S. A. Kale, "Effect of Leading Edge Radius and Blending Distance from," *International Journal of Energy and Power Engineering,* vol. 4, no. 5-1, pp. 54-58, 2015.

[8] J. Lim, "Application of Parametric Airfoil Design for Rotor Performance Improvement," 2022.

[9] M. Hazewinkel, "Encyclopaedia of Mathematics: Supplement Vol. 1.," *Springer Science & Business Media,* p. 119, 1997.

[10] S. Bernstein, "Démonstration du théoréme de Weierstrass fond´ee sur le calcul des probabilités," in *Comm. Kharkov Math,* 1912.

[11] B. M. Kulfan, "Universal Parametric Geometry Representation Method," *AIAA,* 2008.

[12] J. E. Williams and S. R. Vukelich, "The USAF Stability and Control Digital DATCOM. Volume I. Users Manual.," 1979.

[13] "Description of Digital Datcom," USAF, [Online]. Available: https://www.pdas.com/datcomDescription.html.

[14] M. Drela, "XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils," in *Conference on Low Reynolds Number Airfoil Aerodynamics*, University of Notre Dame, 1989.

[15] NASA, "Vortex-lattice utilization," NASA-Langley, Washington, 1976.

[16] S. Suresh, S. Omkar, V. Mani and T. G. Prakash, "Lift coefficient prediction at high angle of attack using recurrent neural network," *Aerospace Science and Technology,* vol. 7, 2003.

[17] M. C. d. Santos, B. S. d. Mattos and R. d. M. Girardi, "Aerodynamic Coefficient Prediction of Airfoils Using Neural Networks," in *46th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 2008.

[18] K. Balla, R. Sevilla, O. Hassan and K. Morgan, "An application of neural networks to the prediction of aerodynamic coefficients of aerofoils and wings," *Applied Mathematical Modelling,* pp. 456-479, 2021.

[19] Xiaosong Du, P. He and J. R. Martins, "Rapid airfoil design optimization via neural networks-based parameterization and surrogate modeling," *Aerospace Science and Technology,* vol. 113, 2021.

[20] X. Jin, P. Cheng, W.-L. Chen and H. Li, "Prediction model of velocity field around circular cylinder over various Reynolds numbers by fusion convolutional neural networks based on pressure on the cylinder.," *Physics of Fluids,,* vol. 30, no. 4, 2018.

[21] H. Chen, L. He and W. Q. a. S. Wang, "Multiple Aerodynamic Coefficient Prediction of Airfoils Using a Convolutional Neural Network," *Symmetry,* 2020.

[22] C. Duru, H. Alemdar and Ö. U. Baran, "CNNFOIL: convolutional encoder decoder modeling for pressure fields around airfoils," *Neural Comput & Applic,* vol. 33, p. 6835–6849, 2021.

[23] C. DURU, H. ALEMDAR and Ö. U. BARAN., "A deep learning approach for the transonic flow field predictions around airfoils," *Computers and Fluids,* vol. 236, 2022.

[24] Y. Zhang, W. J. Sung and D. N. Mavris, "Application of Convolutional Neural Network to Predict Airfoil Lift Coefficient," *AIAA 2018-1903. 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018.

[25] P. R. Spalart and S. R. Allmaras, "A One-Equation Turbulence Model for Aerodynamic Flows," *AIAA Paper,* vol. 92, 1992.

[26] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference for Learning Representations*, San Diego, 2015.

[27] E. F. Toro, "HLLC Riemann solver," in *ath School on Numerical Solutions of Partial Differential Equations,*, Malaga, Spain, 2010.

[28] F. R. Menter, "Two-equation eddy-viscosity turbulence models for engineering applications," *AIAA Journal,* vol. 32, no. 8, pp. 1598-1605, 1994.

[29] C. Rumsey, "2D NACA 4412 Airfoil Trailing Edge Separation Validation Case," NASA, [Online]. Available:

https://turbmodels.larc.nasa.gov/naca4412sep_val_sst.html. [Accessed 11 2021].

[30] P. Cook, M. McDonald and M. Firmin, "Aerofoil RAE 2822 - Pressure Distributions, and Boundary Layer and Wake Measurements," Experimental Data Base for Computer Program Assessment, 1979.

[31] C. D. Harris, «Aerodynamic Characteristics of a 14 Percent Thick NASA Supercritical Airfoil Designed for a Normal Force Coefficient of 0.7,» *NASA-TM-X-72712,* 1975.

[32] AGARD Fluid Dynamics Panel, «Experimental Data Base For Computer Program Asessment,» Technical Editing and Reproduction Ltd Harford House, London, 1979.

APPENDICES

A. Model Usage

The model takes 3 inputs: the airfoil in Selig format, angle of attack list, and Mach number list.

The output of the model consist of 4 columns: Mach number, angle of attack, drag coefficient and lift coefficient. An example can be seen on Table 9.1.

Table 9.1 Example output of the model

Ma	**AoA**	**CD**	**CL**
0.64	-2	0.00858	-0.11599
0.64	e	0.00863	-0.09722
0.64	-1.6	0.00858	-0.05758
0.64	-1.4	0.00852	-0.02684
...	...	...	...

An additional output for drag divergence Mach number can also be chosen. It returns an array of lift coefficient vs $\mathrm{Ma}_{\mathrm{DD}}$.

Table 9.2 Model output example

CL	$\mathbf{Ma_{DD}}$
-0.100	0.776
-0.058	0.785
-0.016	0.789
0.026	0.794
...	...

B. CST Script with Python

Table 9.3 Python script for CST parametrization

```
import numpy as np

from numpy import math

fact = math.factorial

from numpy.linalg import inv


def CSTOLS(upper_true, order, zTe):
    zU = upper_true[:,1] - upper_true[:,0]*zTe
    Bp = np.zeros((order+2, len(zU)))
    x = upper_true[:,0]
    for k in range(order+1):
        C = (fact(order)/fact(k)/fact(order-k))
        Bp[k,:] = np.sqrt(x)*(1-x)*C*(x**k)*((1-x)**(order-k))

    Bp[order+1,:] = x*np.sqrt(1-x)*(1-x)**order

    Bp = Bp.T

    Bu_plus = np.matmul((inv(np.matmul(Bp.T,Bp))),Bp.T)
    vU = np.matmul(Bu_plus,zU)
    return vU

# Inputs: 2D x&y array, CST order, trailing edge thickness

# Output: CST parameters
```